Optical Tracking for Relative Positioning
in Automated Aerial Refueling

THESIS

James H. Spencer, Major, USAF

AFIT/GE/ENG/07-22

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

AFIT/GE/ENG/07-22

OPTICAL TRACKING FOR RELATIVE POSITIONING
IN AUTOMATED AERIAL REFUELING

THESIS

Presented to the Faculty

Department of Electrical and Computer Engineering

Graduate School of Engineering and Management

Air Force Institute of Technology

Air University

Air Education and Training Command

In Partial Fulfillment of the Requirements for the

Degree of Master of Science in Electrical Engineering

James H. Spencer, B.S.E.E.

Major, USAF

March 2007

AFIT/GE/ENG/07-22

Optical Tracking for Relative Positioning in Automated Aerial Refueling

James H. Spencer, B.S.E.E.

Major, USAF

Approved:

/signed/	12 Mar 2007
Lt Col M.E. Goda, PhD (Chairman)	date
/signed/	12 Mar 2007
Dr. S.C. Gustafson (Member)	date
/signed/	12 Mar 2007
Dr. J.F. Raquet (Member)	date
/signed/	12 Mar 2007
Maj B.A. Kish, PhD (Member)	date

AFIT/GE/ENG/07-22

Abstract

An algorithm is designed to extract features from video of an air refueling tanker for use in determining the precise relative position of a receiver aircraft. The algorithm is based on receiving a known estimate of the tanker aircraft position and attitude. The algorithm then uses a known feature model of the tanker to predict the location of the features on a video frame. The features, both structural and painted corners, are extracted from the video using a corner detector. The measured corners are then associated with known features and tracked from frame to frame. For each frame, the associated features are used to calculate three dimensional pointing vectors to the features of the tanker. These vectors are passed to a navigation algorithm which uses extended Kalman filters, and data-linked INS data to solve for the relative position of the tanker.

The algorithms are tested using data from a flight test accomplished by the USAF Test Pilot School using a C-12C as a simulated tanker and a Learjet LJ-24 as the simulated receiver. This thesis describes the results and analysis of the vision system. The algorithm works in simulation using real world video and TSPI data. The system is able to provide at least a dozen useful measurements per frame, with and without projection error. Estimation of features on the tanker in the image is the dominant source of error in the design. The mean feature detection error was 2.7 pixels for the 12.5mm lens and 1.95 pixels for the 25mm lens with a clear background and accurate navigation updates. This level of accuracy should be useful to the navigation system in determining the relative position of the tanker aircraft. The vision system design is heavily dependent on the accuracy of the navigation updates. It is not robust enough to handle situations where the navigation update is considerably inaccurate.

Acknowledgements

The work in this thesis would not have been possible without the help of many people. First I would like to thank my thesis advisor, Lt Col Goda, for his guidance throughout the thesis process and allowing me the latitude in decision making along the way. I would also like to thank the committee members Dr Raquet, Dr Gustafson, and Maj Kish for taking time from their busy schedules to review my work and contribute their thoughts to this research effort.

I would also like to thank the TPS Project Cyclops team; Lieutenant John Bush, Captain Justin Hsia, Flight Lieutenant Karl Kinsler, Captain David Petrucci, and Captain Eric Rucker. They literally made the flight test happen. In addition, I would like to thank the instruction and experience of Mr Russ Easter of Calspan and Mr Bob Wright of Boeing. Their help was crucial to the success of the program.

Finally, I would like to thank my wife and children for enduring my schedule and workload. Their love and patience is truly a blessing.

James H. Spencer

Table of Contents

List of Figures

List of Tables

x

List of Symbols

List of Abbreviations

Optical Tracking for Relative Positioning in Automated Aerial Refueling

I. Introduction

The United States military has increasingly used unmanned aerial vehicles (UAV) to enhance its capabilities in combat. UAVs are traditionally used in intelligence, surveillance, and reconnaissance (ISR) roles. Their role has been evolving to replace many of the manned systems in the defense inventory. The trend is toward combat UAVs capable of lethal strike missions. Programs such as the former Joint Unmanned Combat Air Systems (J-UCAS) program are making ambitious leaps in technology and capabilities beyond the first generation and contemporary UAVs. The future of UAVs involves weaponized air vehicles with network-centric architecture and distributed command and control. The future UAV must be inter-operable with manned and unmanned platforms for collaborative operations. Future UAVs will be employed globally and will require increased range and endurance over previous UAVs. An air refueling capability will make this possible.

One of the longstanding advantages of UAVs is their ability to loiter for extended periods of time. Their primary limitations are fuel. Human factors, such as fatigue, are not issues. Multiple UAV crews can operate the vehicle on shifts. The UAV advantage of lengthy flight time is also its limitation. With a fixed quantity of fuel, there is a trade-off between range and on-station times. One tried-and-true method of increasing range and loiter time is air refueling. The added capability to air refuel increases flight time and range nearly indefinitely.

1.1 Air Refueling Methods

There are two basic methods of air refueling in use today. The U.S. Air Force uses a "flying boom" to refuel its fixed wing aircraft in flight. The boom is a rigid telescoping tube with aerodynamic control surfaces. An operator on the tanker air-

craft flies the boom into a receptacle on the receiver aircraft while the receiver flies formation in the contact position. Fuel is pumped into the receiver aircraft at up to 6000 pounds per minute. The boom refueling method is illustrated in Figure 1.1.

Figure 1.1:   KC-135R refueling an F-22 using the boom refueling method.

The second method of air refueling is the probe-and-drogue method used by the U.S. Navy. A flexible hose with a stabilizing basket at the end, known as a drogue, trails behind the tanker aircraft. A receiver aircraft is equipped with a probe that must be flown into the drogue to enable fuel transfer. Fuel is transferred at a maximum rate of approximately 2000 pounds per minute. The probe-and-drogue refueling method is shown in Figure 1.2.

1.2   Automated Air Refueling Problem Statement

Air refueling is inherently dangerous due to the close proximity of aircraft. Interactions with manned tanker aircraft leave no room for error or miscalculations. To achieve an automated air refueling capability, the Air Force Research Laboratory seeks to develop a combination of Global Positioning System (GPS), inertial and

Figure 1.2: A modified Boeing 707 operated by Omega refuels an F/A-18 using the probe-and-drogue method.

vision sensors to achieve highly reliable and highly accurate relative position sensing required for successful automated aerial refueling operations.

1.3 Machine Vision

The use of a vision sensor to estimate the tanker-UAV relative position vector has several advantages. First, the vision sensor is passive, requiring no emissions that can be detected, jammed, or spoofed in a combat environment. Second, the sensor requires no modifications to the tanker, which would be cumbersome due to the high modification costs.

One challenge in using a vision sensor for AAR is to estimate the relative position of a tanker aircraft from an electro-optic (EO) sensor mounted in the receiver aircraft. The method investigated in this thesis involves identifying points of interest in the video of the tanker and calculating three-dimensional vectors to these points in the camera frame. These vectors can be passed to a navigation integration system for the final relative position determination. The system design is tightly coupled with the navigation system in that it does not compute an optical-based position and attitude

solution prior to integration with the inertial measurements. The navigation system, which is not the subject of this thesis, can use the feature measurements directly.

1.4 AAR History and Related Research

The *Unmanned Aircraft Systems Roadmap* [17]lists automated air refueling (AAR) as a desired future capability for unmanned aircraft systems (UAS)[1] and outlines future funding to develop the capability. The Air Force published a vision document titled *The U.S. Air Force Remotely Piloted Aircraft and Unmanned Aerial Vehicle Strategic Vision* [7]. The strategic vision summarizes the advantages of AAR for UAVs. These advantages includes an increased range and endurance, a reduced number of aircraft deployed, and a reduced need for forward-deployed support. The strategic vision also states:

> ...UAVs must be pre-positioned or self-deployable to be operationally relevant in a rapidly-developing situation. Air refueling capability is essential for larger systems.

The main Air Force AAR research effort began in support of the J-UCAS program. While the Defense Advanced Research Projects Agency (DARPA) managed the J-UCAS program, they outlined the capabilities of the J-UCAS:

> The J-UCAS program is a joint DARPA-Air Force-Navy effort to demonstrate the technical feasibility, military utility and operational value of a networked system of high performance, weaponized unmanned air vehicles to effectively and affordably prosecute 21st century combat missions, including Suppression of Enemy Air Defenses (SEAD); Electronic Attack (EA); precision strike; surveillance/reconnaissance; and persistent global attack within the emerging global command and control architecture. The operational focus of this system is on those combat situations and environments that involve deep, denied enemy territory and the requirement for a survivable, persisting combat presence. [4]

---

[1]The Office of the Secretary of Defense document uses a different terminology. It refers to unmanned aircraft (UA) rather than UAV and uses UAS to include the ground and support elements. The UAS terminology was also adopted by the Federal Aviation Administration.

Although the 2006 Quadrennial Defense Review cancelled the Air Force involvement in the J-UCAS program, the Air Force continues research to develop an air refueling capability for UAVs. The capabilities of the J-UCAS will be applied to future UAVs and could also be implemented in the prospective long-range strike aircraft program. The effort currently underway to enable AAR includes developing a hybrid navigation system that incorporates a vision sensor with differential GPS (DGPS).

Why is the EO sensor being investigated? The primary reason for adding an EO sensor is for dissimilar redundancy, since there is the possibility of losing the GPS solution due to jamming or other malfunctions. Both current and developmental UAVs rely on GPS for navigation. However, their operations are intended to be single-ship missions with no direct interaction with manned aircraft. The precision GPS solution does not perform well, if at all, in a jamming environment. The EO sensor provides a low-cost referee capability to the DGPS system, which has a lower technical risk, but much higher system cost. The EO research will determine the degree of accuracy of EO subsystems in relative positioning. The subsystems may enable positioning as accurate as the precision GPS system.

The future of UAV operations will require formations of UAVs as well as interoperability with manned aircraft (e.g., air refueling). These requirements are being driven from top levels of the Air Force and Air Combat Command (ACC). ACC is seeking the ability to operate UAVs using fighter-style operations such as formations packages and fighter refueling procedures (i.e., different from bomber or heavy aircraft refueling procedures).

Several other sensors are considered. One is a mm-wave radar. However this sensor is prohibitive due to receiver integration issues such as the size of the sensor. Conceivably, the power of the emitter could be reduced so as to not be detrimental to the mission. However, there is still a problem making the sensor conformal and integrating it in a low-observable way. An infra-red (IR) sensor is being considered

and has many advantages. It is able to 'see through' weather and other conditions which give electro-optic sensors difficulty. The problem using it for this thesis was not in concept but in the reality of putting together the flight tests. The EO camera was readily available and adaptable to existing aircraft. There is some question as to whether the funding for the IR sensor would be possible on the desired time-line. The IR sensor is still being researched for future development. In fact, the Air Force Research Laboratory (AFRL) sponsored a flight test in September 2006 to gather IR data.

Why use only a passive sensor rather than an emitter of some sort? One factor is the desire to keep the receiver vehicle low-observable at all times. Another factor is that most active illuminators would also illuminate the boom operator of the tanker at close range. There is potential for health risks unless the proper emitter and power combination are used. The addition of emitters or sensors on the tanker (other than as required for the data link) is not desired.

1.4.1 AFRL Research. ARFL sponsored a flight test done in September, 2004 which collected EO data as well as GPS/INS data. The data were collected from a camera mounted in a Learjet LJ-25 acting as a surrogate UAV while performing simulated refueling with a KC-135R. Using the video as well as precision GPS and correlated data from an inertial navigation system (INS), Boeing analyzed the accuracy of an EO positioning algorithm.

The three dimensional position and orientation of the tanker aircraft, or pose, was estimated using a pose algorithm which compared measured locations in the camera frame and adjusted for windscreen warping, with a surveyed data-base of the tanker aircraft on the ground [3]. The pose method was based on DeMenthon's work [6], which combines two algorithms. The first algorithm estimates the pose from orthography and scaling, and the second algorithm iterates the pose estimates.

The results of their analysis showed an approximately $1m$ difference between the camera-to-boom joint distance (as measured by the video). These differences were

within the uncertainties of the positions of the INS, GPS antennas, and camera. There was a consistent bias between the camera vector positions which was attributed to incomplete field calibration of the camera field of view (FOV). The calibration targets only covered the center area of the FOV, and there was a lack of targets on the outer half of the field of view. These positions were extrapolated from the given targets in the center of the FOV. In addition to the calibration problems, the pose algorithm had difficulties when the feature points were not symmetrically distributed (as occurred with sun glare from the right side of the aircraft). The solution had significant jumps in the range estimates when features exited the camera FOV.

1.4.2 VisNav. Researchers at Texas A&M developed a vision-based navigation system for autonomous air refueling called VisNav [23]. The VisNav system was primarily developed for probe-and-drogue refueling, which is standard for the U.S. Navy. VisNav uses a set of light-emitting diodes (LED) mounted on the drogue which emit structured, modulated light. The light is modulated with a waveform that makes each LED, or beacon, easily distinguishable from the other LEDs. The receiver aircraft is equipped with a position-sensing diode which measures electric currents produced by the LEDs. The navigation solution is then calculated using a Gaussian least-squares differential-correction (GLSDC) routine. The system also requires a feedback loop to adjust the gain of the LED output. This feedback is accomplished via an IR optical or radio signal. The VisNav system could also be used for a boom refueling system by placing the beacons on the receiver aircraft and the sensor on the tanker [8].

The VisNav system developed by Texas A&M has been considered by AFRL but to this point is not being actively pursued. The concept appears to works well, but has several drawbacks. Several modifications to the tanker are required for boom refueling. One objective of the AAR program is to add an AAR capability without tanker modifications. In addition, adding *any* active emitters is undesirable. Another drawback is that the vision sensor would not be capable of aiding formation flight

from the observation position off the tanker. There is also no mention of failure states in which one or more of the LEDs is inoperable. Finally, it has to be 'ruggedized'. Finding a manufacturer to produce and harden the sensor to make it airworthy is a noteworthy obstacle.

1.4.3 Visual Pressure Snake Optical Sensor. Another approach to boom refueling proposed by Doebler, et al, is to use a vision based realtive navigation system which controls the boom of the tanker aircraft while the receiver (UAV) maintains position with GPS [8]. The system uses a visual pressure snakes optical sensor integrated with an automatic boom controller. The visual snake is a closed, non-intersecting contour which is iterated and tracked across images. The target is a geometric pattern painted on the receiver aircraft in the vicinity of the refueling receptacle. The sensor on board the tanker tracks the receptacle and feeds the control system which steers the boom to contact. This system pertains only to operations within the refueling envelope and does not address the rendezvous, closure to the refueling envelope, or the station-keeping method of the receiver in the refueling envelope.

1.4.4 University of Pisa. Research conducted by the University of Pisa [18] suggests a system similar to the VisNav system. The proposed AAR navigation method estimates position based on the localization of infrared markers which have a known geometric distribution over the tanker body or drogue. The difference between this method and the VisNav approach is that the Pisa method does not require active optical markers with modulated light. It also works with passive, undistinguishable markers. Similar to the methods in this thesis, it uses a general framework that includes feature extraction, feature matching, and matching validation. The features, IR LEDs, are arranged across the tankers body or drogue such that at normal refueling positions and attitudes they form a non-intersecting polygon. The IR LEDs form bright spots in the image which are filtered from the background. All the pixels associated with a single LED are grouped, and the detected features are then matched. The matching is based on the arrangement of the LEDs. Then the validation and

pose estimation are combined into one iterative module. The pose estimation is accomplished using the LHM algorithm developed by Lu, Hager, and Mjolsness [14]. The algorithms are simulated in Matlab/Simulink® with an experimental setup that includes a model P-51 with 5 LEDs on a robotic arm, and a webcam. This research has not been applied in flight test or analyzed with existing lighting on current Air Force tankers. It is likely that tanker modifications are required.

1.4.5 *NASA AAR Demonstration.* On August 30th, 2006, a NASA F/A-18 conducted the first autonomous air refueling engagement [5]. The project was a joint effort between DARPA and the NASA Dryden Flight Research Center. The probe-and-drogue engagement was accomplished using the Autonomous Airborne Refueling Demonstration (AARD) system. The AARD system uses GPS-based relative navigation coupled with an optical tracker. The system was developed by the Sierra Nevada Corporation with the optical tracking system provided by OCTEC Ltd.

The video tracking algorithms were cued by a relative GPS/INS. The acquisition function automatically detected the drogue at 60-120 feet for aligning the tracking algorithms [10]. The drogue was required to be detected by 70 to 80 feet. The successful acquisition used a GPS based polar-coordinate estimate of the drogue position and a multiple target tracking algorithm. The acquired drogue information was then passed to a tracking algorithm which continually measured the drogue position and updated the state of the target. The tracker used *a priori* model of the drogue with a shape-finding algorithm. The vision system updated the relative GPS system with the estimated azimuth, elevation, and range of the drogue. Like the design in this thesis, the tracking was accomplished in image coordinate space and transferred to more meaningful units at the output stage.

The system began blending the optical data with GPS data at 80-120 feet and used primarily optical data to complete the contact. Once the drogue was engaged, the vision system was used only to verify the contact status. Although the algorithm

contains many elements common the the design of this thesis, the AARD methods do not translate well to boom refueling.

1.4.6   West Virginia University Research.   At West Virginia University, thesis research was accomplished that investigated methods similar to those in this thesis. Vendra [24] researched the use of corner detection algorithms for UAV AAR. The work was primarily targeted at a comparison between two corner detection algorithms, the Harris and SUSAN algorithms. The algorithms were simulated using a Simulink®-based simulation environment. A simulated image was captured from a virtual reality environment in Simulink®. The image was processed with a corner detector to determine corner locations. The locations of the true corners were assumed to be known as well as the position orientation vector which contained the relative Euler angles. The detected corners were then matched to the true corners with simple correlation of points. Given that the true location was known at a precise instant in time, the matching was nearly trivial. This data was then passed to a pose estimation algorithm. Similar to the Pisa study, Vendra also used the LHM pose estimation algorithm.

1.5   Thesis Overview

Chapter 1 introduces the AAR navigation problem and provides a selected background of research accomplished to develop AAR technologies. Similar methods have been proposed, although all vary slightly. Some disadvantages of each are presented. In Chapter 2, a general background of the reference frames, feature extraction methods, and the tracking system elements are described. Chapter 3 lays out the vision system design and describes the flight test on which the analysis is based. Chapter 4 examines the vision system performance in terms of feature detection performance and tracking performance. These elements are examined in the context of a baseline refueling profile, some environmental factors, and a different lens. Finally, the last chapter explains the conclusions and suggestions for future research.

II. Background

Some crucial elements of the vision system developed in this thesis are the coordinate systems, the camera model, feature detection method, and the tracking system. This chapter gives background information necessary for understanding the vision system design. Three reference frames are defined: the aircraft (model) frame, the body frame, and the camera frame. Next the camera model is developed, followed by an explanation of the feature detection function. Finally, the elements of the tracking systems are described.

2.1 Reference Frames

Three basic reference frames are of interest in this thesis. All reference frames are illustrated with respect to the two test aircraft, a USAF C-12C and a Learjet LJ-24. Further description of the flight test program is given in Section 3.2.

The first reference frame of interest is the aircraft frame of the tanker (C-12C), which is used for the tanker feature model. The second is the body frame, which is used for both the tanker and receiver aircraft by the navigation system. The last reference frame of interest is the camera frame on the LJ-24 receiver aircraft.

2.1.1 Aircraft (Model) Frame. The aircraft (model) frame is a body-fixed frame similar to the coordinate system often used by aircraft manufacturers. In this system, points are defined by fuselage station (FS), buttock line (BL), and waterline (WL). The coordinate system for the model is in the same orientation with the same origin. The x axis is positive toward the tail of the aircraft. The y axis is positive from the center of the aircraft toward the right wing. The z axis is positive toward the top of the aircraft. The $x - z$ plane is the aircraft plane of symmetry. The origin is located in front (14.2 inches for the C-12) and below the aircraft (87 inches below the rear door hinge). The notation used in this thesis for a point in the the aircraft frame is $[x_{at}, y_{at}, z_{at}]$. Figure 2.1 shows the aircraft (model) frame.
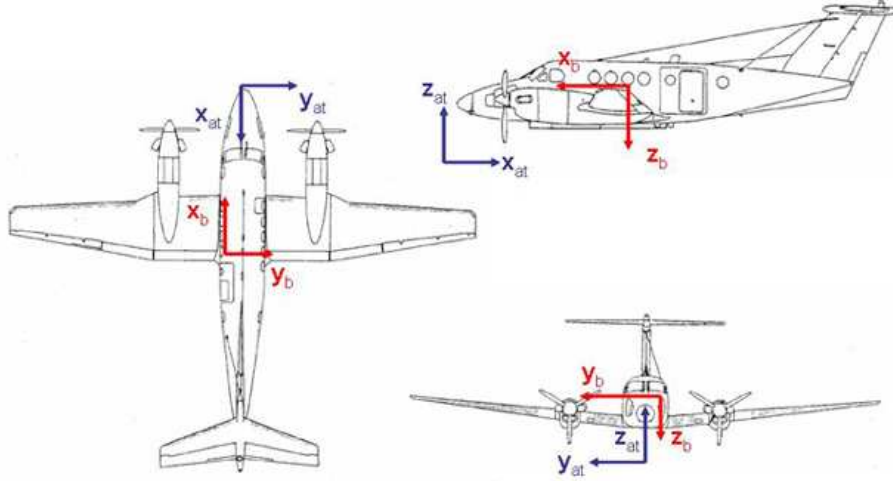
Figure 2.1:   The aircraft (model) frame and the body frame are shown
for the C-12.

2.1.2   Body Frame.   The body frame is used for both the tanker and re-
ceiver aircraft. The origin and orientation are fixed with respect to the geometry of
the aircraft. The origin used for both aircraft is at the INS computational center.
Typically, a body-fixed frame has the origin at the center of gravity (CG). In this
case it coincides with the INS for convenience. The x axis is positive toward the nose
of the aircraft. The y axis is positive toward the right wing. The z axis is positive
toward the bottom of the aircraft. The $x - z$ plane is parallel to the aircraft plane of
symmetry. The body frame is shown in Figures 2.1 and 2.2 for the C-12 and LJ-24
respectively. The receiver body frame is used in this thesis as the 'world' reference
frame for the camera model.

2.1.3   Camera Frame.   The camera coordinate frame has the origin at the
center of the image plane. The z axis is the optical axis perpendicular to the image
plane in which the lens center lies. The lens center lies at coordinate $(0, 0, f)$, where
f is the effective focal length. The y axis is through the top of the image plane, and
the x axis goes through the left side of the image plane. The camera for the flight test
was installed on the glareshield of the LJ-24 as shown in Figure 2.2. A more detailed

12
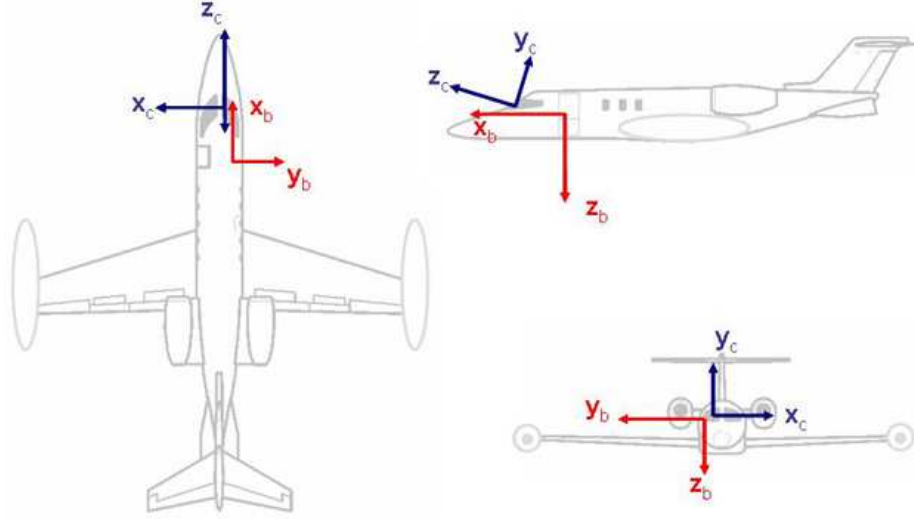
Figure 2.2:   The body and camera reference frames are shown for the LJ-24.

illustration of the camera frame is shown in Figure 2.3 with its relationship to the digital image.
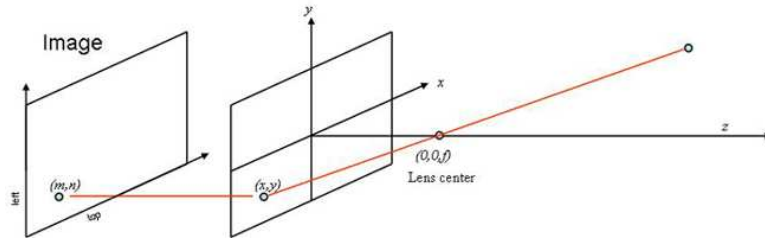
Figure 2.3:   The camera coordinate frame with its relationship to the digital image.

2.2   Camera Model

The camera model projects the 3-D aircraft feature model onto the 2-D image in several steps using the pinhole camera model described by Gonzalez [9]. The basic objective is to obtain the image-plane coordinates of a point viewed by the camera. The camera model applies a set of transformations which first aligns the camera and world coordinate systems, then a perspective transformation is applied. The situation is shown in Figure 2.4.

13
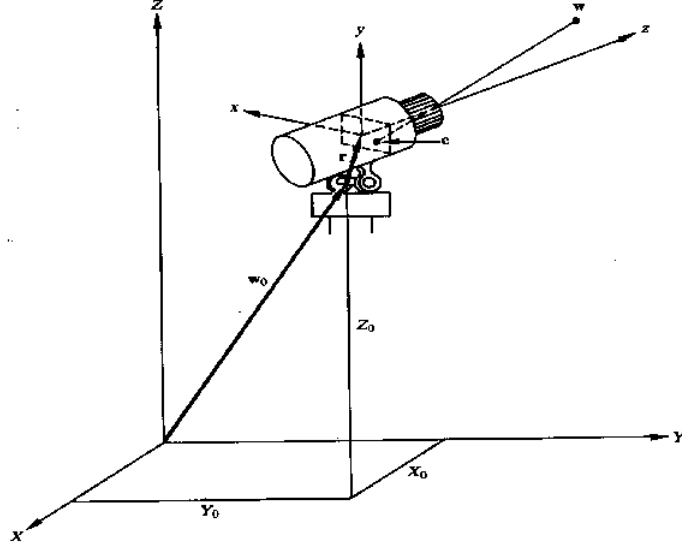
Figure 2.4:   The camera model geometry [9].

A point $\mathbf{x}_w$ is located in the world coordinate system $[X\ Y\ Z]$. To allow the transformations to occur in a linear algebra context, the cartesian coordinates are changed from $[X\ Y\ Z]^T$ to the homogeneous form $\mathbf{x}_w = [X\ Y\ Z\ 1]^T$.

The camera, with a different coordinate system (x, y, z), is offset from the world coordinate frame by a constant vector $\mathbf{w}_o$. This vector denotes the location of the camera gimbal, which allows an angular pan ρ and tilt τ. The offset from the gimbal to the image plane is represented by vector $\mathbf{r}$.

The first step in accomplishing the projection of world point $\mathbf{x}_w$ is to apply transformations that align the world and camera coordinate systems. First, the world point coordinates are adjusted by applying the displacement of the gimbal center from the world origin, which is done by applying matrix G shown below. The operation $G\mathbf{x}_w$ translates the origin of the world point to the gimbal center.

$$G = \begin{bmatrix} 1 & 0 & 0 & -X_o \\ 0 & 1 & 0 & -Y_o \\ 0 & 0 & 1 & -Z_o \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

14

The points of the model are then rotated through the pan and tilt angles. The pan angle is defined as the angle between the x and X axes. The tilt angle is defined as the angle between the z and Z axes. A combined rotation matrix R serves to rotate the world coordinate system to align it with the camera coordinate system:

$$R = \begin{bmatrix} cos(\rho) & sin(\rho) & 0 & 0 \\ -sin(\rho)cos(\tau) & cos(\rho)cos(\tau) & sin(\tau) & 0 \\ sin(\rho)sin(\tau) & -cos(\rho)sin(\tau) & cos(\tau) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

The final transformation to align the camera and world coordinate systems is to translate the origin from the gimbal center to the image plane by vector $\mathbf{r}$ with components (r_1, r_2, r_3). Here, as shown in Figure 2.3, C translates the points to the image origin which lies in the center of the image plane:

$$C = \begin{bmatrix} 1 & 0 & 0 & -r_1 \\ 0 & 1 & 0 & -r_2 \\ 0 & 0 & 1 & -r_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

The three dimensional world point $\mathbf{x}_w$ is in the camera frame after applying the series of transformations $CRG\mathbf{x}_w$. Finally, the perspective transformation is accomplished with the use of the projection matrix P:

$$P = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & \frac{-1}{f} & 1 \end{bmatrix}.$$

This matrix projects the world point onto the image plane using a mathematical approximation of the image formation process. After putting the transformations

together, as shown in Equation (2.1), the result is the camera coordinates in homogeneous form $\mathbf{x}_h$. The cartesian coordinates $\mathbf{x}_c$ are extracted by dividing by the fourth element of $\mathbf{x}_h$. The third component of $\mathbf{x}_h$ is of no interest. In fact, all z information is lost in the transformation. The projection has no inverse without retaining or having prior knowledge of the z information of the world point that created the image point. The combined tranformations are

$$
\mathbf{x}_h = PCRG\mathbf{x}_w = \begin{bmatrix} x_h \\ y_h \\ z_h \\ \frac{-z_h}{f} + 1 \end{bmatrix} \Rightarrow \mathbf{x}_c = \begin{bmatrix} \frac{fx_h}{f - z_h} \\ \frac{fy_h}{f - z_h} \end{bmatrix}. \tag{2.1}
$$

Below is a summary of the key transformations of the pinhole camera model used for image creation:

- $\mathbf{x}_w = [X \ Y \ Z \ 1]^T$ is the feature location in world coordinates

- G translates from world origin to gimbal center

- R rotates through the pan (ρ) & tilt angle (τ)

- C translates to the camera frame origin

- P is the projection matrix

- $\mathbf{x}_c$ is the resulting feature location on image plane

2.3 Feature extraction

Feature extraction deals with the detection, location, and representation of image features corresponding to interesting elements of a scene. Image features, as described by Trucco [22], "are local, meaningful, detectable parts of an image." Local properties could be points, lines, curves, shape features, textures, or structures of gray-levels. Global features, such as the average gray level, are also used in computer

16

vision but are not discussed in detail in this thesis. Feature extraction is a necessary step in computer vision, not the objective.

Corner detection is commonly used in computer vision because it has many application areas, for example: motion tracking, stereo matching, and image database retrieval. In a gray scale image corners are areas where the image gradients in two orthogonal directions are high. More importantly, corners are *discrete*, meaningful points that are detectable. Edges in images are also useful in many applications. Edges are defined where the image gradient is high in one direction and low in the orthogonal direction. An edge detector finds meaningful features; however, because they are not discrete they are not easily and explicitly trackable. Corner features are more easily characterized and are explicitly trackable in image sequences.

Several corner detection algorithms have been developed, and a few are mentioned here. Moravec [15] pioneered work in 'interest points'. His corner detector examined small changes in image intensity when shifting a local window in various directions. Harris [11] addressed the limitations of Moravec's work and applied corrective measures to enhance the algorithm. Smith and Brady developed a new algorithm known as the SUSAN (Smallest Univalue Segment Assimilating Nucleus) corner detector [20]. The SUSAN detector operates based on a brightness comparison in a circular mask. Shen and Wang recently developed a corner detector that uses a modified Hough transform to organize edge lines and detect corners [19].

The Harris corner detector was selected for this thesis based on a comparison with a Shen/Wang corner detector. This comparison found that the Shen/Wang corner detector is more accurate in localizing corners; however, the detector had significant problems with acute angled corners as well as real images.

2.3.1 Harris Corner Detector.   The Harris corner detector, also known as the Plessy corner detector, finds corners locally by shifting a window and measuring the changes in image intensity [11]. Corners in an image are located where two lines or edges intersect. The edges may be formed by one or more objects. These lines

17

or edges are indicated by image intensity gradients. Corners occur where the image gradients are high in two orthogonal directions.

The shifting window is represented by the change in intensity E in Equation (2.2). Harris expanded the equation using gradients X and Y:

$$
\begin{aligned}
E(x,y) &= \sum_{u,v} w(u,v) \left[I(x+u, y+v) - I(u,v) \right]^2 \\
&= \sum_{u,v} w(u,v) \left[xX + yY + O(x^2, y^2) \right]^2.
\end{aligned}
\tag{2.2}
$$

The gradients X and Y are defined in Equation (2.3) and are approximated by convolving the image I with a derivative mask of $[-1, 0, 1]$ and $[-1, 0, 1]^T$ for X and Y, respectively:

$$
\begin{aligned}
\mathsf{X} &= \left(\tfrac{\delta I}{\delta x} (x_o, y_o) \right) \\
\mathsf{Y} &= \left(\tfrac{\delta I}{\delta y} (x_o, y_o) \right).
\end{aligned}
\tag{2.3}
$$

For small shifts, the change in intensity E can now be written

$$
E(x,y) = Ax^2 + 2Cxy + By^2
\tag{2.4}
$$

where

$$
\begin{aligned}
A &= \mathsf{X}^2 \otimes w \\
B &= \mathsf{Y}^2 \otimes w \\
C &= \mathsf{XY} \otimes w.
\end{aligned}
\tag{2.5}
$$

These gradients are smoothed with a circular Gaussian window w to reduce noise effects, and the gradients are placed into a gradient density matrix:

$$
M = \begin{bmatrix} A & C \\ C & B \end{bmatrix},
\tag{2.6}
$$

18

which follows from the fact that E from Equation (2.4) is

$$E(x, y) = (x, y)M(x, y)^T.$$

The gradient density matrix M is real and symmetric, so it can be diagonalized with orthonormal eigenvectors which 'rotate' the matrix to its principal orthogonal axes. The eigenvalues give the gradient magnitudes in the rotated frame. By comparing the eigenvalues of the gradient density matrix, the nature of the pixel can be determined as follows:

- If both eigenvalues are large, it is a corner.

- If both are small, it is a flat region.

- If one is small and the other is large, it is an edge.

The gradient density matrix exists for each pixel in the image. Instead of explicit eigenvalue decomposition for each M, which would be costly computationally, Harris uses the determinant $(AB - C^2)$ and the trace $(A + B)$ to develop a 'corner measure'. The corner measure R_c is a scalar measure for each pixel which is compared to a user defined threshold to determine if the pixel contains a candidate corner:

$$R_c = det(M) \ - \ k \ tr(M)^2.$$

The corner measure includes an empirical constant k which is normally 0.04 - 0.06. The threshold for the corner measure R_c is a critical parameter. If the threshold is set too low, there are an excessive number of false corners. If the threshold is set too high, the detector can miss true corners in the image. The threshold is also image dependant and is normally set manually in literature. Thus there is a set of corner pixels for which R_c lies above the threshold. These candidate corners are then filtered to find the local maxima of the corner measure, which labeled a corner.

One of the disadvantages of the Harris corner detector is that there are several parameters which the user must specify. These include the variance σ^2 of the Gaussian smoothing window, the threshold of the corner measure, the empirical constant k, and the radius for non-maximal suppression (which finds the strongest corner in each local neighborhood). A different corner measure is suggested by Noble [16] which eliminates the constant k and uses an arbitrarily small positive number, ϵ.

$$R_c = \frac{det(M)}{tr(M) + \epsilon}$$

Figure 2.5 shows the results of the Harris corner detector using the Noble corner measure. The green dots indicate the detected corners.
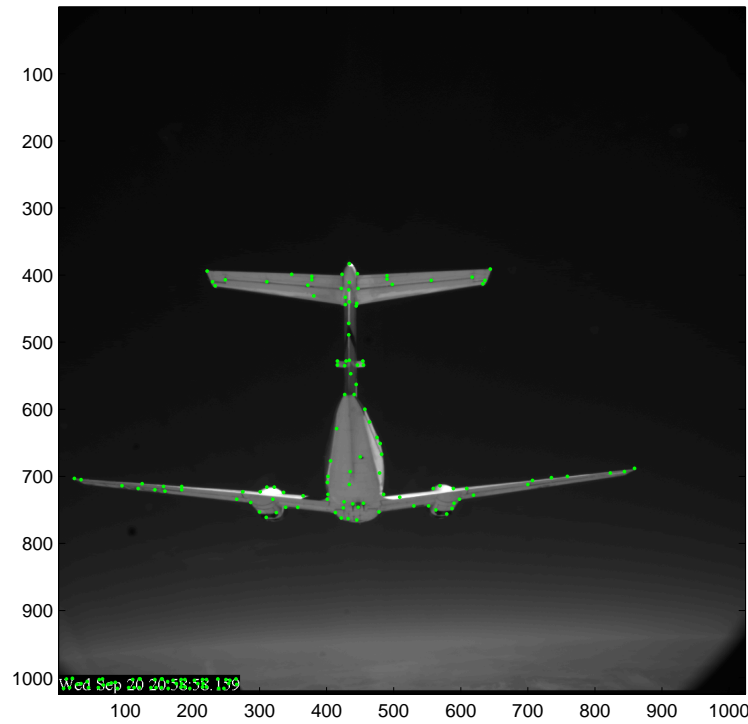
Figure 2.5:   A C-12C is shown with the Harris corner detector applied. The green dots indicate the detected corners.

20

2.4 Tracking System Elements

In a sequence of images such as from a camera on an aircraft in AAR, the motion of an observed scene is nearly continuous if the sampling time is small enough. Because of this continuity, features in the scene are predictable based on their previous trajectories. Because of the number of features being tracked, much of the design shown later is this thesis contains elements of a multiple-target tracking (MTT) system. A typical MTT system requires a sensor to detect potential targets of interest, algorithms to initiate and delete tracks, associate measurements to tracks, and filters to estimate and predict specific target parameters [2]. Examples of target parameters are target position and velocity.

2.4.1 Measurement Processing.   The measurement processing in this thesis consists of the images taken from the digital camera combined with the feature extraction algorithm. An EO sensor operates in the visible portion of the electromagnetic spectrum. The energy detected is primarily produced by light reflected from objects in the scene.

The image is then scanned by the corner detector. The detected corners are the observations that once associated become measurements. In a more general sense, an observation is the term used to refer to all observed quantities included in a detection output, such as kinematic parameters like position. An observation should also include an estimate of the time observed. Observations generally occur at regular intervals such as data frames. In this case, the image is equivalent to one scan, and the scan rate is approximately 30 frames per second. The only target quantities measured directly are the positions of the corners within the image.

Processing the detected corners involves logic to account for several issues. Included in the observations are false corners as well as corners of interest. Also like other sensors, the corner detection has a limited useful resolution. Thus, it cannot detect corners too close together unless the masking is set so low that a excessive number of false corners are detected.

2.4.2 Data Association. The overall function of the data association segment is gating, observation-to-track associations, and track maintenance [2]. A MTT system requires a complex data association logic in order to sort out the sensor data into targets of interest and false signals. There is no standard approach for all applications. The tracking designer must choose based on knowledge and experience the technique which is best suited to his application.

2.4.2.1 Gating. A gating technique is used to determine which observations (detected corners) are candidates to update existing tracks. It is a screening mechanism that limits the number of association calculations performed by eliminating unlikely pairings. Gating is done based on estimates of the current location of the tracks. In this case the tracks are the features from the tanker model. Incoming observations are checked to see if they are "reasonable" for observation to track pairing. The gate is essentially a criterion, such as a window, which allows a number of observations to pass through for consideration to update a track. A representation for a simple gating criteria is

$$d_{ij} \leq G_i, \tag{2.7}$$

where d_{ij} is the Euclidean distance of observation j from track i and G_i is the gate size or threshold. If observation j meets the gating criteria, it is kept and considered a candidate for observation-to-track pairing.

Some potential issues arise which must be handled by the association logic. For closely spaced targets, a single observation may be produced. Also, the gates of closely spaced targets may overlap. In many cases, more than one observation lies within a track gate. Conversely, one observation may lie within the gates of more than one track. These problems are depicted in Figure 2.6. Each of these issues is addressed through the formation of an assignment matrix in the association logic.

Choosing a gating technique involves several considerations. The gate size is based on a maximum allowable error along with the statistics of the targets and sensor. For instance, the statistics of how the target moves is important. Tracking a

22
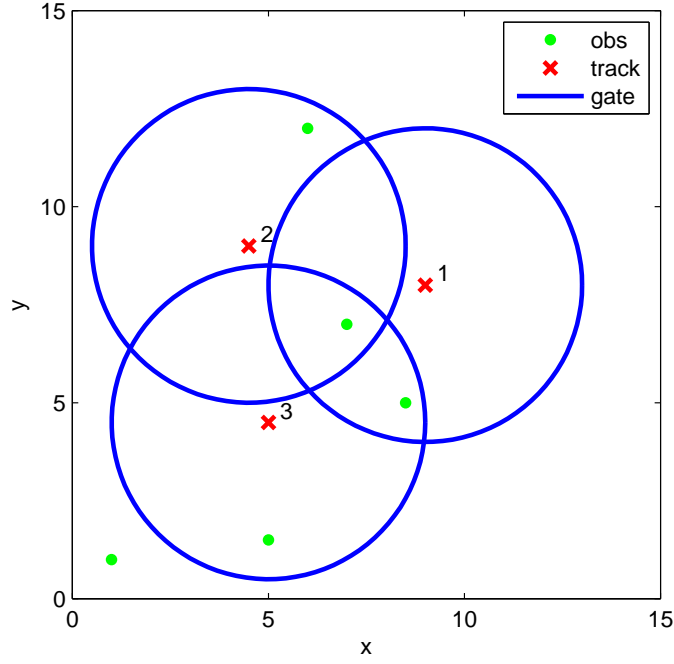
Figure 2.6: Some typical gating and data association issues are shown. These include, more than one observation in a single gate and overlapping gates which contain shared observations. The tracks depicted are the estimates of the current track location.

high performance jet with radar requires gating based on the dynamic capabilities of the aircraft. However, tracking a large truck on a highway is a different problem. In the case of air refueling with a receiver-mounted camera, the motion of the receiver is primarily along the optical axis of the camera. Objects in the camera typically move radially from a focus of expansion (FOE). Movement is slower at longer ranges and the feature velocities increase as range is decreased. In addition, features accelerate as they reach the edges of the image.

2.4.2.2 Association. The association function takes the paired observations and tracks that satisfy the gating criteria and determine which observations actually update each track. The most widely used and straightforward method for association is the global nearest neighbor (GNN) algorithm [2]. This method considers all candidate observations for all tracks and assigns unique observation-to-track

pairings such that at most one observation is paired with a single track. Two observations cannot be assigned to a single track. Likewise, two tracks cannot share a single observation. The pairing is usually done to minimize a cost function or maximize likelihood.

GNN is a unique-neighbor approach. In contrast, other algorithms such as a simple nearest neighbor may use the closest observation to each track for the update. A situation may occur that allows one observation to update more than one track if it is the closest observation to both tracks. Multiple hypothesis testing (MHT), like GNN, is considered a unique neighbor approach; however, it uses multiple scans to determine pairings [2]. This is referred to as deferred logic, and it allows the pairings to be postponed until more information is available from more data frames.

This thesis focuses on the GNN algorithm, which uses a sequential logic in which only one frame determines observation-to-track pairings (a single hypothesis). It attempts to find and propagate the single most likely association hypothesis to the next frame. The algorithm seeks the maximum number of assignments with the minimum total cost. GNN also allows computation of track scores which can be used for track maintenance.

As seen in Figure 2.6, the association algorithm must resolve conflicts such as overlapping gates with shared observations and multiple observations within a single gate, which is done through the use of an assignment matrix.

The assignment matrix is developed based on likelihoods. One general approach forms the elements of the assignment matrix based on a score

$$a_{ij} = G_i - d_{ij}, \tag{2.8}$$

where a_{ij} is the score associated with assigning observation j to track i and G_i is the gate associated with track i. In the two dimensional case, the score is the margin by which observation j clears gate G_i. An example of the generalized assignment matrix is shown in Table 2.1.

24

Table 2.1:   Example Generalized Assignment Matrix.

	O1	O2	O3	Om
T1	a_{11}	a_{12}	a_{13}	a_{1m}
T2	a_{21}	a_{22}	a_{23}	a_{2m}
Tn	a_{n1}	a_{n2}	a_{n3}	a_{nm}

Instead of minimizing cost, the objective of the matching algorithm is to maximize the gain. The gating problem is transparent at this point because only observations which pass the gating criteria are considered. The matrix shown in Table 2.1 does not consider adding new tracks.

There are several algorithms that can produce solutions to the assignment matrix. The algorithm chosen for this thesis is the auction algorithm [2]. The auction algorithm seeks to maximize the total gain and also finds the maximum number of assignments. The solution is an iterative process of bidding and assignments. In the bidding phase, each observation bids for its best track based on the current score a_{ij} (cash available) and the track price P_i. After the observation finds the best track, it bids for the track and raises the track price. The track price P_i is increased by the difference between the best and second best assignment values for observation j, which is done so that the observation is able to 'buy' its second-best track if another track steals its best assignment. Any other observation bid on that track is unassigned so that it can bid in another round.

The algorithm, as given by [2], is shown below.

1. Initialize all observations as unassigned. Initialize track prices P_i to zero.

2. Select an observation j that is unassigned. If none exists, done.

3. Find the "best" track i_j for each observation j: Find i_j such that

$$a_{i_j j} - P_{ij} = \mathbf{max}_{i=1,...n}(a_{ij} - P_i)$$

4. Unassign the observation previously assigned to i_j (if any) and assign track i_j to observation j.

5. Set the price of track i_j to the level at which observation j is almost satisfied.

$$P_{ij} = P_{ij} + y_j + \epsilon$$

where y_j is the difference between the best and second best assignment values for observation j and ϵ is a small raise in price.

6. Return to step 2.

The uniqueness of this algorithm stems from the fact that each observation considers its second-best pairing each time it bids on a track. The iterative process allows tracks to be stolen back and forth until the solution converges. An important note on ϵ is that the value must be sufficiently small so that the solution converges to the same result regardless of the order of the observations. An ϵ that is too small wastes time because more iterations are required. A large ϵ allows the solution to converge more quickly, but the solution depends on the ordering of the observations. The auction solution to the gating and data association example shown in Figure 2.6 is shown below in Figure 2.7

Sufficiently spaced tracks improve data association due to non-overlapping gates. There are essentially no secondary matches for an observation to consider. Since there are typically more observations than tracks, there is an increased chance of divergent paths. In this case, paths can wander without being deleted.

2.4.2.3 Track Maintenance.   Track maintenance refers to the functions of track initiation, confirmation, and deletion. Track initiation logic can be accomplished in several ways. Two methods are initiate tracks for all unmatched observations, or use multiple hypothesis testing (MHT). Track initiation based on all unmatched observations is a simple method but can lead to a large number of spurious tracks. The preferred method of MHT starts tentative tracks based on unmatched
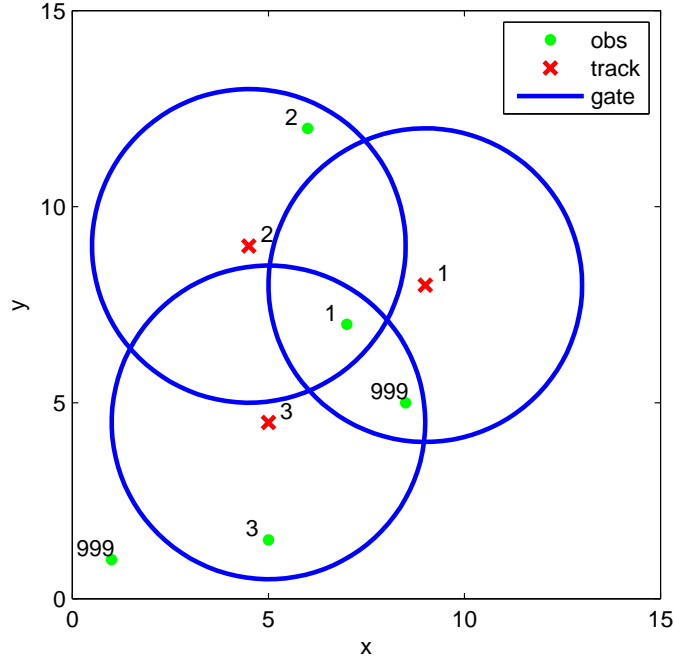
Figure 2.7: The result of the auction algorithm for solving the data association issues shown in Figure 2.6. The number 999 designates an unmatched observation.

observations, then uses subsequent data to determine which of the tentative tracks are valid [2]. The validity of a track is increased each time it is associated with an observation. Track confirmation is required because of the high probability of spurious single observations. The gate size and number of observations required to confirm a track are a functions of the confidence in the validity of the original observation. A typical method of confirmation requires M associated observations within N scans.

Track deletion occurs when a track becomes starved of observations. A track becomes degraded when it is not updated frequently. If a significant amount of time passes without an update, or if the track is of low quality, the track is deleted. A typical rule may be that a track is deleted if there are no observations within N scans. A more useful method is to use a track score that reflects the quality and frequency of updates. If the track score is not within tolerance, the track is deleted.

The details of other track scores can be found in [2]. The specific methods used for track maintenance in this design are outlined in Section 3.1.5.

2.4.3 Filtering and Prediction.   The purpose of the filtering and prediction components is simply to estimate the track parameters at the current time and provide a prediction of the track parameters at the next sample time. These predictions are extremely important because they form the basis of the gates to be established at the next sample time. Modern tracking systems typically use Kalman filter models for updating and propagating track parameters. Two simpler filters are examined in this thesis. The first is a zero-order hold (ZOH) and the second is an α-β filter. The α-β filter is a fixed-coefficient filter and is much simpler than the Kalman filter which computes filter gains dynamically along with a covariance for the state vector being estimated. (The α-β filter does not compute a covariance for the states.) The ZOH and α-β filters will be described in the sections that follow.

2.4.3.1 Zero-order Hold Filter.   The ZOH filter is the simplest possible filter. It is examined as a baseline filter in accordance with the philosophy of Occam's razor: the simplest solution which works "well enough" is the best solution. The ZOH essentially makes no assumptions about the dynamics of the target or noise in the sensor [1]. The ZOH filter also tends to work in low dynamic situations, which is a valid assumption for much of the tracking for AAR. Thus, the track parameters (in this case position) are exactly where they are observed, and they are most likely to appear in the same place in the next frame.

For comparison, the ZOH is presented with a measurement model, update equation, and propagation equation. Let the parameters of interest be the row and column of the feature in the image. The state vector $\mathbf{x}(k)$ is

$$\mathbf{x}(k) = \left[\begin{array}{c} r(k) \\ c(k) \end{array} \right]$$

28

where k indicates a discrete sample time. The noiseless measurement model

$$z(k) = H\mathbf{x}(k) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \mathbf{x}(k), \tag{2.9}$$

where $z(k)$ is the measurement state vector.

The update equation (although trivial) states that the target is estimated to be exactly where it was measured. The current estimate of the state vector $\mathbf{x}(k)$ given all information included in the scan at time k is denoted $\hat{x}(k)^+$ or $\hat{x}(k|k)$:

$$\hat{x}(k)^+ = z(k). \tag{2.10}$$

The propagation equation is shown in Equation (2.11). It states that the target is expected to be in the same spot at the next sample time. The estimate of the state vector at the next time $k + 1$ is $\hat{x}(k + 1)^-$ or equivalently $\hat{x}(k + 1|k)$:

$$\hat{x}(k + 1)^- = x(k)^+. \tag{2.11}$$

2.4.3.2 Alpha-Beta Filter. The α-β filter is a fixed coefficient filter with a very simple implementation. It can be used when only position measurements are available [1]. A fixed-coefficient filter also has a computational advantage when a large number of targets are present. The α-β filter is considered a noiseless dynamics constant velocity filter.

The measurement model for the $\alpha - \beta$ filter is the same as the ZOH filter. The state vector $x(k)$ now includes the velocities of the feature decoupled into row and column.

$$z(k) = Hx(k) \tag{2.12}$$

where

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}, \quad \hat{x}(k) = \begin{bmatrix} \hat{r}(k) \\ \hat{c}(k) \\ \hat{v}_r(k) \\ \hat{v}_c(k) \end{bmatrix}$$

The update equation, now implements the fixed gains, α for position, and β for velocity. The second term contains a weighting matrix W and the residual $[z(k) - H\hat{x}(k)^-]$. The residual term is the difference between the measurement state vector $z(k)$ and the prediction of the measurement $H\hat{x}(k)^-$. The update equation is

$$\hat{x}(k)^+ = \hat{x}(k)^- + W \left[z(k) - H\hat{x}(k)^- \right], \tag{2.13}$$

where

$$W = \begin{bmatrix} \alpha & 0 \\ 0 & \alpha \\ \beta/T & 0 \\ 0 & \beta/T \end{bmatrix}. \tag{2.14}$$

The weighting matrix W uses the fixed gains to update the position by α times the residual. It updates the velocity estimate by β/T times the residual of the velocity where T is the sample period. The noise terms are not directly used in the propagation and update equations, however, the measurement and process noise are implicit in the constant gains.

Given the updated estimate from the current time, the estimate for the next sample time is given in Equation (2.15). The position estimate is the current position plus the velocity times the sample period. The estimate of the velocity is assumed essentially constant over the period of propagation. The predicted state, $\hat{x}(k+1)^-$ is found by multiplying the updated estimate by the state transition matrix F:

30

$$\hat{x}(k+1)^- = F\hat{x}(k)^+, \qquad (2.15)$$

where

$$F = \begin{bmatrix} 1 & 0 & T & 0 \\ 0 & 1 & 0 & T \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \qquad (2.16)$$

Some important things to notice are that when $\alpha = 1$ and $\beta = 0$, the above simplifies to a ZOH filter. Both the α-β and ZOH tracker hypothesize a constant velocity, so error is increased for target accelerations. If the motion is low-dynamic in nature and the sampling time is quick enough, the ZOH filter works adequately. The ZOH is sensitive to target velocity. The α-β filter is more robust than the ZOH for handling target accelerations, and is less sensitive to target velocity.

III.   Design and Flight Test Overview

In this chapter the design of the vision system is explained followed by descriptions of each subsystem. These functions include image point estimation, image segmentation, feature extraction, and tracking. A short explanation of the error sources is also given. The second section contains information about the flight test conducted at Edwards AFB by the USAF Test Pilot School.

3.1   Design Overview

The design of the vision system is similar to many other designs in that it involves feature extraction, feature matching, matching validation, and ultimately pose estimation (although pose estimation is not examined directly in this thesis). The overall design, including the navigation system, is similar to a tightly coupled GPS/INS in which the INS computes the navigation parameters using the pseudorange measurements directly, instead of having the GPS compute a navigation solution and then combining that with the INS navigation solution. In this design the navigation system directly incorporates measurements of the feature locations. The feature measurements are treated similarly to pseudoranges (although the measurements are fundamentally different). The interaction of the vision system with the navigation system is shown in the top right of Figure 3.1. The vision system is a subsystem of the coupled design.

The navigation system initializes the vision system with an estimate of the tanker relative location and the Euler angles. The vision system uses this information along with the tanker model to predict the locations of the tanker features in the image. The image is then segmented into sub-images which contain clusters of predicted features. The feature detector uses corner detection in each sub-image to detect features and match them with predicted features. The measurements are converted and passed to the navigation system for pose estimation. The matched features are also tracked from frame to frame to enable continued detection for various navigation updates or dropouts.
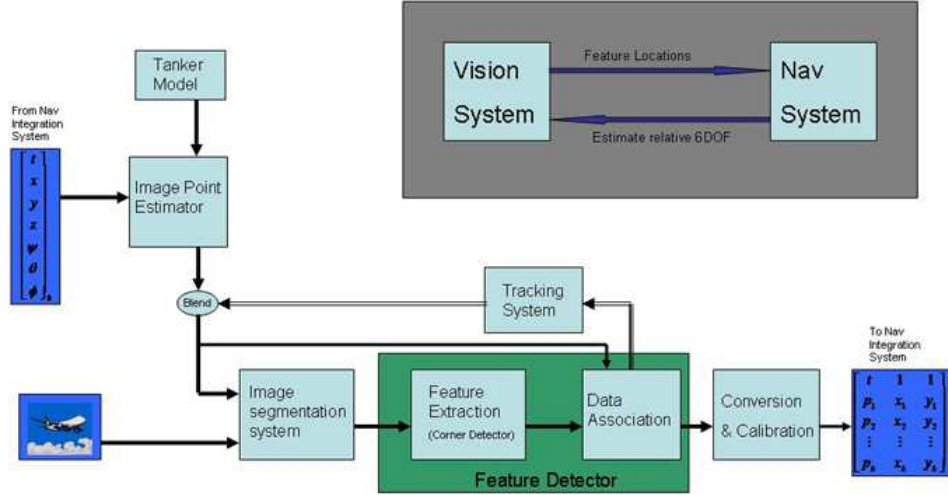
Figure 3.1:   The interaction of the optical system with the navigation system is shown in the upper right corner. The design contains feature extraction with validation and tracking.

With previous algorithms, such as the Vendra's work [24] and DARPA's AARD [10], there is a transition from GPS in the rendezvous to machine vision in the final refueling area. The design considered in this thesis fuses machine vision and differential GPS/INS throughout the entire air refueling.

Some underlying assumptions are made in creating this design. First, the video is assumed to be monocular, which is valid for the flight test done by AFRL [3] and the testing done by the USAF Test Pilot School (TPS) [21]. It also is applicable to a failure state in a stereo vision application. A stereo vision application makes the 3-D location of features much easier. Secondly, only ranges inside 500 feet are considered. Outside 500 feet, the corner detection is of little use (in this application) and other feature extraction methods are required. Thirdly, the work assumes flight near or within the nominal air refueling envelope (and not in the observation position). Specifically the receiver is assumed to be behind and below the tanker aircraft. The initial conditions of the vision system are assumed to be such that there is a fairly accurate initial estimate of the tanker position and orientation. Initializing the vision system with EO-only methods is not addressed. Also assumed is that the video sampling rates, and the navigation update rates are not constant, which is one of

33

the major reasons for sensor-level tracking. Finally, air refueling is assumed to be a low-dynamic environment. Azimuth, elevation, and closure rates of a receiver are normally low for safety reasons, which validates this assumption.

3.1.1 Image Point Estimation. The objective of the image point estimation routine is to create location estimates of 3-D features in the 2-D image plane. The image point estimator uses the estimated relative six degrees-of-freedom from the navigation integration system along with the tanker model to generate estimates of where the model points are located in the image, which is done in several stages. First, the relative position vector,

$$\hat{X}_b = [t \ x \ y \ z \ \psi \ \theta \ \phi]^T \,, \tag{3.1}$$

which containes the estimated *relative* position (x, y, z) and attitude (ψ, θ, ϕ) of the tanker in the body frame of the receiver aircraft, is converted to the camera frame. The vector also contains the time t associated with the vector.

This vector is then used to rotate and translate the tanker model in the camera frame. Next, a camera model was used to project the 3-D model onto a 2-D image. Finally, the projected points are calibrated using a calibration model to correct for windscreen warping in the receiver.

3.1.1.1 Tanker Model. The tanker model for the C-12C was created by the Cyclops test team [21] and is described in more detail in Section 3.2.2. The tanker model contains 29 measured feature locations. It was created using a surveyed area with multiple manual measurements. Figure A.1 shows a picture of the C-12C from a typical refueling viewpoint along with the measured features for the tanker model. The feature descriptions can also be found in Table A.1.

34

3.1.1.2   Conversion to the Camera Frame.   The tanker model is first rotated by the relative yaw ψ, pitch θ, and roll ϕ angles with the direction cosine matrix (DCM) [1]

$$
C_m^b = \begin{bmatrix} c(\psi)c(\phi) & s(\theta)s(\psi)c(\phi) - c(\theta)s(\phi) & s(\theta)s(\phi) + c(\theta)s(\psi)c(\phi) \\ c(\psi)s(\phi) & c(\theta)c(\phi) - s(\theta)s(\psi)s(\phi) & c(\theta)s(\psi)s(\phi) - s(\theta)c(\phi) \\ -s(\psi) & s(\theta)c(\psi) & c(\theta)c(\psi) \end{bmatrix} . \qquad (3.2)
$$

The model is then translated to the point where the model origin is at the tip of the $[x, y, z]^T$ vector (extracted from $\hat{X}_b$) in the body frame of the receiver aircraft.

The model points are then correctly represented in the body frame of the receiver aircraft. From this point, the camera model described in Section 2.2 is applied. The camera parameters correspond to the actual Dalsa Pantera TF 1M60 described in Section 3.2.1.1. The body frame of the receiver aircraft is used as the world reference frame described in the camera model. The $\mathbf{w}_o$ and $\mathbf{r}$ vectors are fixed as a function of the camera installation. The camera gimbal is fixed in this case with a pan angle, ρ, of 0° and a tilt angle, τ, of approximately 16°.

The predicted locations of the features in the image plane consist of these projections of the tanker model based on the relative position and orientation provided by the navigation system.

3.1.1.3   Calibration Corrections.   Because the camera is 'looking' through a surface beyond the lens (in this case a windscreen), the image is a distorted version of the real scene. Calibration matrices are used to move the real projections of the features to locations where they would have fallen through the windscreen.

The calibration matrices consist of two matrices. The first matrix characterizes the horizontal distortion in the form of a δ_{row}, and the second matrix gives the vertical

---

[1]The *sin* and *cos* functions are abbreviated s and c to shorten the notation.

distortion in the form of δ_{column}. For instance, a point in space that falls into pixel (r, c) is distorted and appears in pixel $(r + \delta_{row}, c + \delta_{column})$. A sample three-dimensional representation of the calibration matrices is shown in Figure 3.2. The calibration matrices are described in more detail in Section 3.2.3
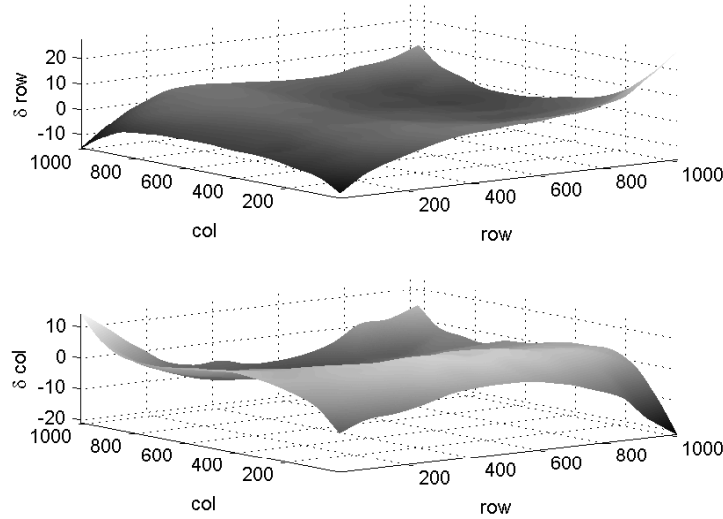
Figure 3.2: Sample camera calibration matrices are shown for the 12mm lens. A feature that normally falls in pixel (r, c) is distorted and appears in pixel $(r+\delta_{row}, c+\delta_{column})$.

3.1.2 Image Segmentation. The purpose of image segmentation is to partition the digital image into disjoint sets of pixels, each of which corresponds to a region of interest, which is done to reduce the processing time per image as well as to reduce the number of false features. In the context of the vision system design, the feature extraction segment dominates processing time, which is primarily a function of image size. Segmenting the image into sub-images containing regions of interest greatly reduces computation time for feature extraction and reduces the number of false features, which in many ways is analogous to aiming the sensor or pre-gating.

The segmentation is based on clusters of predicted features in the image. The method used in this design is a version of agglomerative hierarchical clustering (AHC)

36

[13]. AHC begins by assigning each feature to a separate cluster. These clusters are then successively merged based on their distance from another cluster. In this case, the Euclidean distance between each cluster is compared to a threshold distance. If the distance is less than the threshold, the clusters are merged. The merging of clusters in this algorithm wasis based on a single linkage. Thus, if any member of a cluster is close enough to any member of a neighboring cluster, then the clusters are merged. The clusters are invariant to the order in which the clustering is initialized.

After the clusters are formed, a buffer region around the clusters is added to form the limits of the sub-image. The buffer region is based on feature extraction errors, and it is increased for images in which a local histogram equalization is helpful.

As an example, 14 features are handpicked in the image shown in Figure 3.3. First the entire image is processed with the feature detector. Next the image is segmented and processed with the feature detector. The segmented image containes 11 clusters with only 3.9% of the total image area. The segmented image is processed 89% faster than the detection on the entire image. This savings depends on the buffer region around the clusters. In Figure 3.3, a 30 pixel buffer surrounds each cluster, which is reasonably large compared to detection error. Even for reasonably large buffer sizes, the windowed detection is an order of magnitude faster.

3.1.3 Feature Extraction. The goal of feature extraction in the vision system is to isolate important features in the image which could then be used to understand the scene. An image contains far more information than is needed or can be useful to solve a particular problem. Teaching a computer to do what the mind does instantaneously is a formidable task.

As in several other research efforts, the features are discrete points or markers in a known geometry. Because it is tightly-coupled, this design does not require all features to be visible at any given moment. However, a greater number of detected and validated features increases the accuracy and robustness of the navigation solution.

Percentage of area windowed = 3.9681
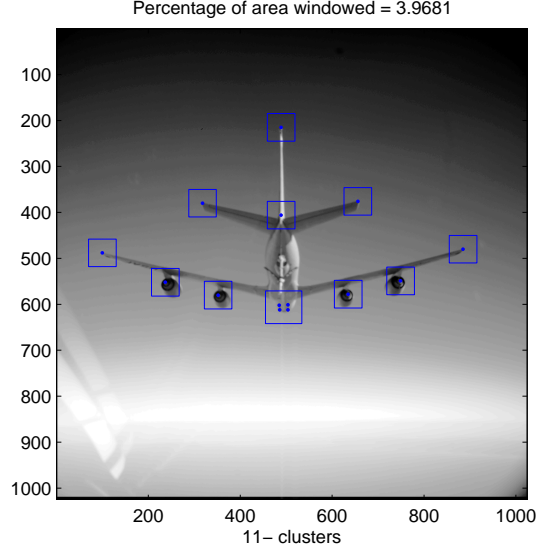
Figure 3.3:   The 1024 by 1024 image is processed with
a corner detector followed by segmenting the image and
applying the corner detector. The segmented processing
is accomplished 93.4% faster.

In some images, such as in twilight conditions, the intensity gradients are too
small due to a small range of gray scale values. In these images, applying a histogram
equalization aided in increasing the contrast and thus the gradients. The main draw-
back is that the noise ratio is also increased and an increased number of false corners
are detected.

The Harris corner detector in Chapter 2 is used with the corner measure devised
by Noble. Another modification made is the use of an automatic threshold based on
the statistics of the corner measure. Since the magnitude of the corner measure
depends on the characteristics in the image, several statistics are tested to find an
adaptive automatic threshold. The best threshold found is the mean of the corner
measure R_c in the sub image. An example of a sub-image is shown in Figure 3.1.3. The
corner measure histogram is shown in the top right plot and resembles an exponential
distribution. Typically, 20-30% of the total R_c for a sub-images exceed the mean of
R_c. After filtering for local maximas, the probability of detecting true corners is high
with few false corners.

38
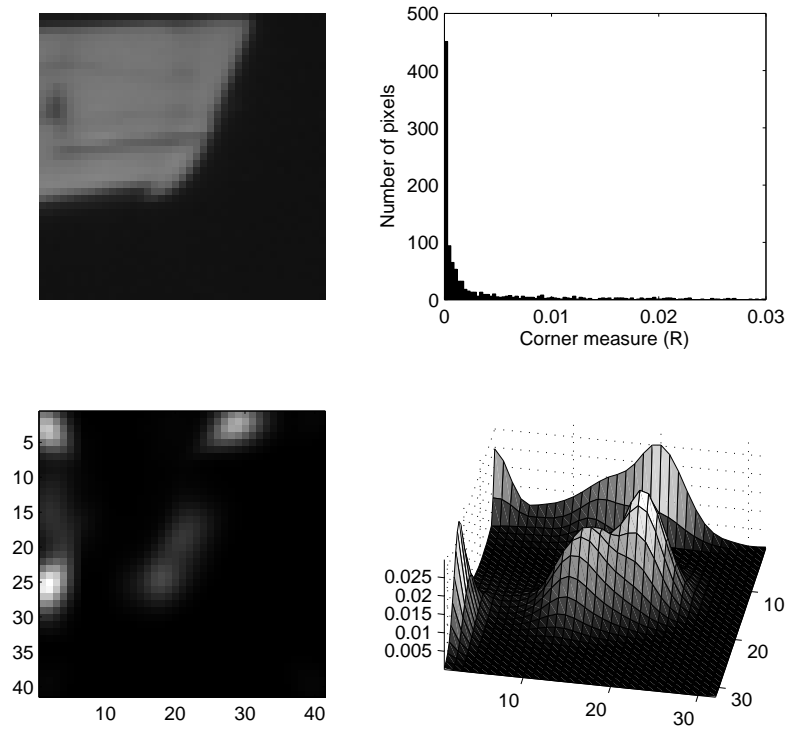
Figure 3.4: The top left image is a sub-image of the right horizontal stabilizer of the C-12. The top right plot shows the histogram of the corner measure which is typical of most of the sub-images. The bottom left shows an image of the corner measure and the bottom right shows the corner measure represented as a 3-D surface.

Another slight modification is made due to the magnitude of the corner measure at the edges and corners of the image. The corner measure blew up near the corners and edges of the image as seen in the bottom left plot in Figure 3.1.3. Thus, the outer 5 pixels around each sub-image are removed from consideration as candidate corners. The bottom right plot in Figure 3.1.3 shows the resulting cropped corner measure image as a 3-D surface.

3.1.4 Data Association. The data association used in the design was based on the global nearest neighbor algorithm described in Section 2.4.2. The gating method uses a Euclidean distance parameter which accounts for measurement er-

rors. This gate value is determined empirically based on the desire to associate tracks that are "relatively" close to observations. The gate is not dynamic, i.e., it does not change size depending on the quality of the track. The GNN method associated each track and observation via an assignment matrix which is solved by the auction algorithm [2].

The data association function assumes that observations received in a single frame contain at most one observation from each target. Also, a single observation may have been the result of two or more closely spaced targets. In this case, the gates formed by the predicted track locations overlap and only one track is updated.

3.1.5 Tracking. The tracking algorithms used here enables feature detection regardless of navigation update rates or dropouts. The tracking accomplished by the vision system is sensor-level tracking, while the navigation system would do central level tracking. With features being tracked on two levels, two questions are, "What is the optimal level of sensor/central level tracking?" and "Is there a need for an optimal tracker at the sensor-level?" The assumption made for this design is that a simplified tracking method is sufficient for sensor-level tracking. Since the navigation system monitors the residuals of each feature, it is able to discard measurements that are misassociated along with diverged tracks.

The detection and tracking are essentially independent in this design. The method is recursive in that the processing is only based on data in the current frame. All previous track information is implicit in the current estimate. There is no explicit batch processing which would process the data for all observations in a moving window of time.

Contrary to normal tracking, track initiation for features is never based on the vision system alone. Tracks are initiated only as a result of incoming pose estimates from the navigation system. An alternate method is to initiate tracks based on unmatched observations. Since these tracks would likely not correspond to features in the tanker model, they are of limited usefulness.

A track score is used during track maintenance and deletion. The track score is based on detections and missed detections. The track score is initialized at zero, and each missed detection raises the track score by one. Each detection reduces the score by one to a minimum score of zero. When the track score reaches an empirical threshold, the track is deleted. During track maintenance, a track that does not have an update from a current observation is simply propagated to the next frame. Unless the track is quickly updated, the track score degrades quickly and the track is deleted.

Certain features are typically stronger features such as a wing-tip. A stronger feature is one in which the corner measure R_c is very high and thus the probability of detection P_D is near one. In addition, the detected corner is well localized to the visible feature in the image. Other features are considered weak, which means that their probability of detection is significantly lower or they are poorly localized. The weaker features are dropped more frequently but are also more likely to wander without being dropped.

3.1.6 Blending. When there is both a current estimate of the feature track from the local sensor-level tracking system and an incoming pose estimate from the navigation system, there is a potential conflict in estimated track location. Using one estimate or the other alone may result in errors, since there is a possibility of both inaccurate navigation updates and inaccurate sensor-level tracking.

The current method of resolving the conflict is to trust the navigation system update. The navigation system normally has higher fidelity information (data-linked INS) and much better filters. For these reasons, it is assumed that the navigation system produces a more accurate estimate of the tanker and thus tanker feature locations.

While using the α-β filter, the feature velocities must also be estimated. Initializing the velocities during a navigation-based update is done by using a previous navigation update while the navigation updates are occurring at a reasonable frequency. During the first navigation update, the velocities are initialized to zero.

The blending method admittedly has the potential to induce errors. These drawbacks can be addressed in future research.

3.1.7   Conversion and Calibration.   The output to the navigation system consists of the detected features (observations that are validated and matched), and the frame time. The estimated locations of the feature-tracks are not sent, only the observed positions. Prior to output of the detected features, the camera calibration discussed earlier is applied in reverse. A feature detected in a certain pixel is moved to where that feature should have been located in the real projection without the windscreen.

The format of the output is

$$
Z(t) = \begin{bmatrix} t & 1 & 1 \\ p_1 & x_1 & y_1 \\ p_2 & x_2 & y_2 \\ \vdots & \vdots & \vdots \\ p_l & x_l & y_l \end{bmatrix},
\tag{3.3}
$$

where $Z(t)$ is the measurement matrix and l is the number of features detected. The first element of the output t is the time of the measurements, and the second and third elements of the first row are arbitrary place holders. The point identification number is given by p_l.

The location of a feature in the camera frame is given by (x_l, y_l) in meters, and it coincides with the location of the center of a pixel. Also of note is that there are a variable number of features per epoch. The navigation system is equipped to handle any number of detected features.

3.1.8   Sources of Error.   There are several sources of error which can limit the performance of the vision system. They are in two categories; measurement origin error and measurement error. Measurement origin error involves those error

sources external to the camera and feature detection. A poor pose estimate from the navigation subsystem causes the vision system to look for features in the wrong place, which is most important during the initialization of the vision system. If the tanker model is in error, the projection and predicted locations of the features are inaccurate as well. A similar error can be introduced with errors in feature tracking estimates.

Another source of measurement origin error is the pinhole camera model. It is a first-order model with limited accuracy for real lens systems. There is also error associated with the camera parameters such as the focal length, camera position, and camera orientation.

Lever arms are measured for the camera, GPS, and INS. Because of the number of coordinate conversions and rotations, small errors incrementally add together to form larger errors. For instance, the lever arm to the camera is easy to measure. However, finding a lever arm to image plane and equivalent focal distance is much more difficult. If the focal plane is in error, or its orientation is in error, several pixels of difference in the predictions versus the actual feature locations can result.

Errors in the windscreen calibration matrices will also introduce errors in the feature measurements. On the front end, there results error in the predicted feature locations from the image point estimation routines. On the tail end, the measured and validated features are de-calibrated for output to the navigation system.

Next there are errors associated with the measurements themselves. There are always errors associated with the sensor, in this case an EO camera. Some examples of this error are bad pixels and saturated pixels. There are several pixels which always have a zero-intensity value associated with them, which are regarded as bad pixels. When a significant part of the image is saturated, a streak in the image occurs as the result of the detectors inability to clear the charge in the (CCD) cells quickly enough between frames.

Another error associated with the corner detector is a localization error. The corner detector used, and most corner detectors (in general), have a localization error

in the detected corner. For example, the corner detector is in error by an average of two pixels on a real image. Feature extraction errors can also be associated with low-lighting conditions, shadows, and obscurement by other objects. Since the observation-track association is considered part of the measurement process, a misassociated feature also introduces error. In this case, the error can lead to a poor feature estimate in the next frame and a divergent track.

3.2 Flight Test Overview

The flight test was conducted by the Cyclops test team at the USAF Test Pilot School. The flight test collected EO, GPS and INS data for use in evaluating optical recognition and tracking algorithms. A USAF C-12C simulated a tanker and a Calspan Learjet LJ-24 simulated an unmanned receiver. Specifically, the aircraft maneuvered between the approach-to-contact, pre-contact, and contact positions at various rates. Data were collected in various environmental conditions such as cloudy background or low light levels. Flight testing was conducted between 11 September and 22 September 2006.

The LJ-24 followed the C-12C in simulated approach-to-contact, pre-contact and contact positions. Relative movement was generated between the aircraft against a variety of sky backgrounds. Most of the testing was performed in day visual meteorological conditions (VMC), except for two sorties which finished at dusk in order to collect low light data.

The overall objective of the test was to gather time-synchronized video and time, space, and position information (TSPI) data in an operationally representative environment. The overall objective had three sub-objectives:

1. Evaluate the utility of the C-12 feature model for use in the optical tracking algorithms.

2. Evaluate the methodology of creating the calibration matrices for the lens effects of the LJ-24 windscreen.

44

3. Gather time-synchronized video and GPS/INS data.

3.2.1 Test Item Description.   The systems under test consisted of GPS Aided Inertial Navigation Reference (GAINR) units that provided time, space, and position information (TSPI) for each airplane, plus a monochrome digital camera mounted on the glare shield of the LJ-24 that provided video data and a recording system. A data link was also fitted that provided the pilots with range information, mainly for safety reasons.

3.2.1.1 EO camera description.   A Dalsa Pantera TF 1M60 monochrome digital camera provided by AFRL was mounted on the glare shield of the trail aircraft (LJ-24) as seen in Figure 3.5. Two different lenses were fitted: a 12.5mm lens for flights 1-5, and a 25mm lens for the final flight. Key characteristics of the camera were: 100% fill-factor, 12-bit digitization, and 1024 by 1024 resolution. The frame rate was approximately 30 frames per second. The camera was controlled through a PC-based system. The video recording system captured optical data through a camera interface card onto two 300GB hard disks. A display of the camera image was available to the test conductor in the rear of the LJ-24.

3.2.1.2 TSPI description.   TSPI was measured and recorded on both aircraft with GPS Aided Inertial Navigation Reference (G-lite) units (configuration C2B). Existing antennae were used for GPS signal collection. Both aircraft also had stand-alone data-link units built by the Air Force Institute of Technology (AFIT) that enabled real-time display of relative position and attitude information in the receiver aircraft. The real-time data-link information was not required for flight but was desired for data quality and added safety.

3.2.2 C-12C Tanker Model.   The C-12 feature model was created using 38 tracking feature points measured from surveyed ground locations around the aircraft. Full details and methodology can be found in the Cyclops Technical Information

Figure 3.5: A Dalsa Pantera TF 1M60 monochrome digital camera provided by AFRL was mounted on the glare shield of the simulated receiver, a Calspan Learjet LJ-24.

Memorandum (TIM) [21]. The uncertainty of the model was expressed as a root mean square (RMS) error. The maximum RMS error for the measured features was 0.27 inches, the minimum was less than 0.005 inches, and the mean RMS error was 0.07 inches.

The model was designed to be robust enough to allow for additional points to be added later in case an optical tracking algorithm was consistently identifying a point that was not a part of the original model. In order to evaluate the methodology, known points from the original model were selected as the new tracking features so that they could be compared to the original model. Relative measurements were taken from the new tracking feature points to other feature points already incorporated into the model, as opposed to the original ground references.

A Euclidean distance was minimized to generate the estimated location of the features in the original coordinate system using physical measurements of the features relative to a subset of the 38 other measured features. The Euclidean distance was minimized using the non-linear optimization function, Solver, in Microsoft Excel,

46

and the uncertainty was expressed as a radial error. Although the measurements of the 'additional' feature points were significantly less accurate than the original measurement taken from surveyed ground locations, the error was within the desired accuracy ($\leq 1\ inch$).

The original feature model was found to be accurate beyond the requirements of the user, and the added feature errors were within tolerances. The methodology was found to be straightforward. Therefore, the model creation method was evaluated as satisfactory.

3.2.3 Windscreen Calibration Matrices. Since the camera was mounted on the glare shield of the LJ-24, the images collected by the camera were distorted by lens effects of the LJ-24 windscreen. As it was envisaged that any future UAV AAR camera would be protected from the airflow by a lens, the test team evaluated a method of creating the necessary calibration matrices to account for the lens distortion.

The camera was calibrated on the ramp by parking the test aircraft at a preset position and using an array of target markers attached to an external hangar wall as shown in Figure 3.6.

The array of target markers filled the entire camera FOV. A baseline calibration was accomplished prior to flight test and, over the flying period, two more calibration events were carried out. The camera and mount were marked with tell-tale paint so that any accidental movement of the system would be detected. With confidence that the camera had not moved, the camera 'looked' through the same calibrated portion of the windscreen and additional calibrations were not required. The positions of the markers and the foreground surface, as marked by tennis balls on paving slab junctions, were measured to create a 3-dimensional model of the array.

During each calibration, several images were captured at up to three positions. The software routine `find_tgts` developed by Boeing SVS [12] then identified the target marker row and column numbers in each image. Based on the estimated camera locations, the routine also projected the target markers onto the camera charge-

Figure 3.6:   The setup of the camera calibration for the flight tests.

coupled device (CCD) array without windscreen distortion. By comparing the projections and the actual images, the calibration software `camera_cal`, also developed by Boeing SVS, then generated two calibration matrices for the LJ-24 windscreen. A set of calibration matrices were created from the images captured at the first location. At the second ground position, the pixel locations of the markers were predicted using these matrices and were compared to the actual image. The difference in pixel locations was then calculated. A second set of calibration matrices was then created using a combination of images from the first and second ground positions. These matrices were then used to predict and compare the pixel locations of the markers at the third ground position. Finally, all three images were used to create the calibration matrices that were to be used in the final post-processing, effectively tripling the number of markers in the array. The methodology of creating the matrices is shown in [21].

The calibration matrices were labeled M_1, M_2 etc (the subscript denotes which images were used in its creation). The differences, or residuals, represent the errors between the different predictions. By comparing the column and row residual means, the matrix estimation software showed a slight positive bias on row correction esti-

mation. The spread on both the column and row residual was on the order of one pixel. With the exception of M_{13}, there was a noticeable improvement in the residuals as the order of the calibration was increased. The order referred to the number of distances used to estimate the composite calibration matrices, e.g., first order used only one distance.

The calibration matrices were accurate to within two pixels for nearly 90% of the pixels in each image. The distortion beyond two pixels was generally located at the edges and corners of the video frames. There were significant differences between the calibration matrices in these locations. It was suspected that this was primarily due to the interpolation and extrapolation errors in the software routine. The results showed that the windscreen distortion was greatest at the corners and edges of the image. The differences between the matrices at two different distances were also the greatest at the edges and corners. Complete results were given in the TPS report [21].

During data reduction, the test team discovered that the calibration software provided by Boeing SVS was very sensitive. Small inaccuracies in the input files drove great variations in the output matrices. To ensure that `camera_cal` correctly generated the calibration matrices, the test team had to examine every target marker location estimated by the `find_tgts` routine, a time consuming and unreliable process.

3.2.4 *In-flight maneuvers and environment.* The in-flight maneuvers were flown to simulate air refueling using the C-12C as a simulated tanker and the LJ-24 as the simulated receiver. The test points included operationally representative refueling as well as maneuvers with increased rates of movement from various positions in and near the refueling envelope.

Three primary positions were defined for the test: contact, pre-contact, and 300 feet in trail. Figure 3.7 illustrates the contact, pre-contact, and 300-feet positions that were used and Table 3.1 provides detailed descriptions of these positions. These positions were slightly modified from the actual refueling envelopes of the KC-135

49

and KC-10. The reference point for the simulated refueling boom was the aft tip of the fuselage. The flight path reference point on the tanker was the intersection of the chord line with the fuselage centerline.
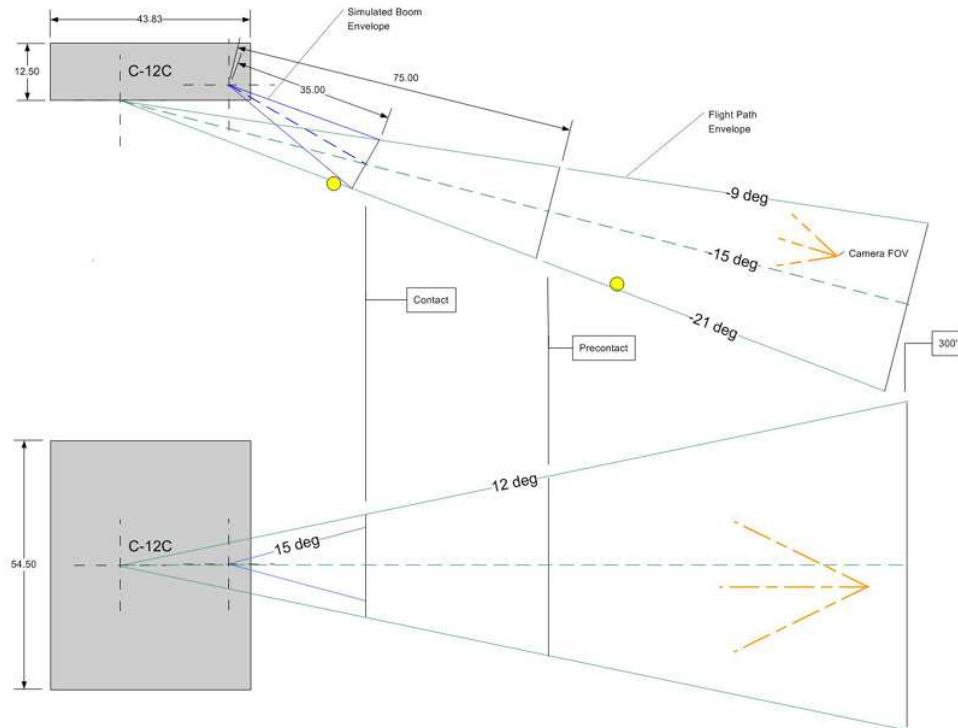
Figure 3.7: The contact, pre-contact, and 300-feet positions that were used are shown.

The test points were divided into five blocks and a brief description of each block is as follows.

- Block 0 - Operationally Representative

- Block 1 - Baseline geometry points (azimuth symmetry)

- Block 2 - Baseline points in turns (bank & azimuth symmetry)

- Block 3 - Higher rates (asymmetric w/r to azimuth)

- Block 4 - Expanded geometry, mixed rates, FOV (asymmetric w/r to azimuth)

Block zero consisted of a closure from 1/2 mile to pre-contact, then contact. The receiver then backed out to pre-contact for one more closure to contact followed

50

Table 3.1:   Refueling Position Desctiptions.

Position	Description
Contact	35 ft nose to tail separation between -9 and -21° elevation within ±12° azimuth
Pre-contact	75 ft nose to tail separation between -9 and -21° elevation within ±12° azimuth
300 feet	300 ft nose to tail separation between -9 and -21° elevation within ±12° azimuth

by an emergency separation. Block one was composed of movements from various positions given by the baseline geometry shown in straight and level flight. These baseline geometric points are shown in Figure 3.8. The movements were made at nominal rates consistent with normal air refueling. Block two mirrored block one except that it was accomplished while in a constant 15° bank. Block three used the same baseline points with increased rates of range, azimuth, and elevation. The final block examined higher rates in more than one parameter, such as a high range and elevation rate, and lateral movement which caused the tanker to exit the camera FOV.
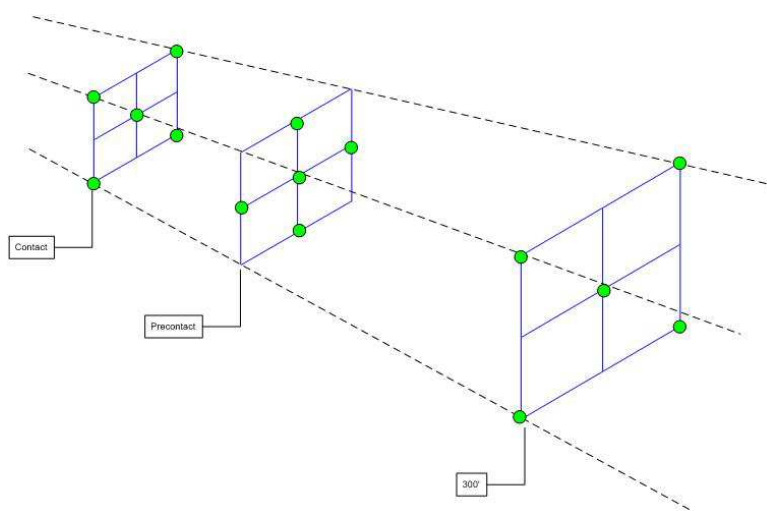

Figure 3.8:   The baseline geometry for test blocks 1,2, and 3.

In addition to geometric rates, other conditions were flown such as high sun angle ($> 45°$ above the horizon), low sun angle ($< 45°$), and twilight. Video was recorded during testing that contained clouds in the background to examine the effects on tracking algorithms. In addition, the last flight was flown with a 25mm lens for comparison with the 12.5mm lens. The camera calibration was repeated for the new lens.

The synchronization of the video time stamps with the GPS/INS data collected by the GAINR-Lite was done by analyzing an in-flight wing-rock maneuver. Both aircraft would accomplish several wing-rock maneuvers per flight by quickly rolling to ± 15 degrees of bank. The GPS/INS recorded the bank angle and time, and the video was analyzed to see when the maximum right bank angle occurred. There was a drift in the time stamps on the video frames when compared to the TSPI data given by the GAINR-Lites. A sample of the time synchronization error is shown in Figure 3.9. For full time synchronization results, see [21].
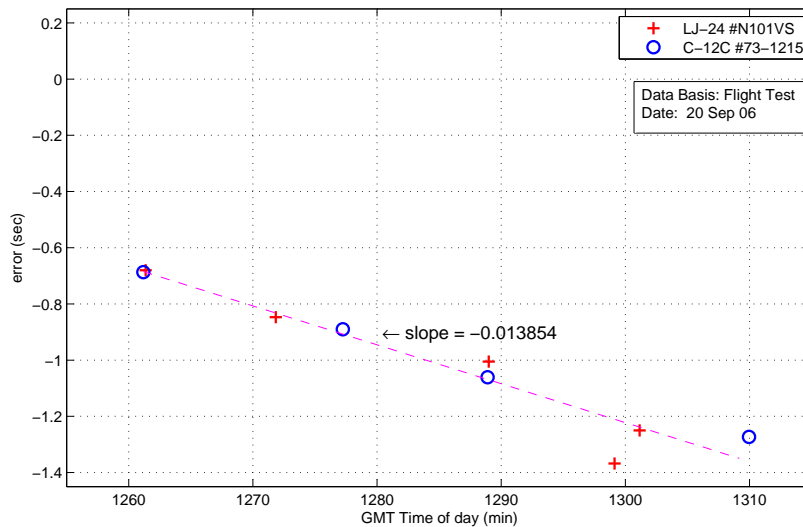

Figure 3.9:   The time synchronization error for the second flight is shown. The error listed is the difference between the maximum bank angle given by the TSPI data and the maximum bank angle apparent in the video segment.

In all, over four hours of data was taken. The next chapter analyzes only a very small portion of the data from the flight test.

IV. Results and Analysis

The errors, strengths, and weaknesses of the vision system are analyzed throughout this chapter in the following areas:

- Image point estimation

- Feature detector performance

- Tracker performance

Those areas are examined in the context of four scenarios. The first scenario is a baseline refueling closure. This closure is representative of a typical refueling profile from 300 feet to the contact position at 35 feet. The second scenario includes a short segment where the receiver aircraft is moving at higher rates of range, azimuth, elevation, and roll. The primary focus of this scenario is the sensor-level tracker performance. The third scenario includes environmental factors such as clouds in the background, low sun angle, and night. The focus here is on feature extraction. The final scenario examines the differences encountered by replacing the 12.5mm lens with a 25mm lens.

The results and analysis in this chapter do not include the closed-loop design of the navigation system. The navigation system for the total system design shown in Figure 3.1 is still under development. Although examination of total system performance is desired, it is to be done at a later date by the developer of the navigation system. As a result, the performance of the vision system is examined.

4.1 Image Point Estimation Issues

Several issues arose from the image point estimation, to include TSPI issues and camera calibration issues. The image point estimation issues regarding the navigation input $\hat{X}_b$ are the result of the TSPI data. The camera calibration issues are not related to the distortion correction matrices ($\delta_{row}, \delta_{col}$). They are primarily the result of an incomplete camera calibration of the camera parameters.

4.1.1 Navigation Data.   The navigation input was provided by the 412 RANS/TSPI office at Edwards AFB. The TSPI data for each aircraft were provided as well as a moving origin reduction for the relative position of the C-12C G-lite from the Lear-24 G-lite. The relative attitude was calculated by subtracting the attitude angles of the C-12C from that of the LJ-24.

The software which provided the moving origin reduction was developed before inertial units were being used for TPSI data. The position data is based solely on GPS and therefore does not account for the attitudes of each aircraft. The reference frame initially given was based on the the flight path vector rather than the body frame of the Learjet. The projection based on this model is shown in Figure 4.1(a).

The relative position vector from the moving origin reduction was rotated through the angles defined by the flight patch vector and the aircraft attitude to compensate for angle of attack, wind drift, and sideslip. The angle of attack and the combination of wind drift and sideslip of the LJ-24 averaged approximately 5° and 6.5°, respectively. The resulting vector provided a significant improvement in the feature projection shown in Figure 4.1(b), however this vector highlighted the remaining issue of the camera parameters.

4.1.2 Camera Parameters.   The camera was installed in a fixed location and orientation. As a result, the camera lever arm from the G-lite and the pan, tilt, and roll angles were constant. One additional parameter of interest was the effective focal length of the lens. The effective focal length was given by the lens documentation. The position and orientation parameters of the camera were measured using the same methods used to boresight the G-lites. The position and orientation parameters were also calculated during the camera calibration for the windscreen distortion. The parameters from the calibration were not referenced to the aircraft orientation, so they were of limited usefulness.

The projection of the feature points after the navigation data were corrected revealed an additional error in the camera parameters. These parameters were mod-
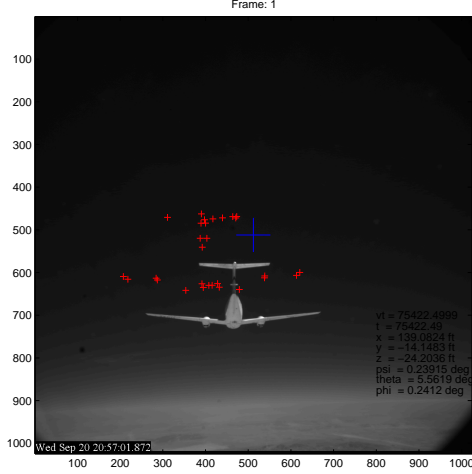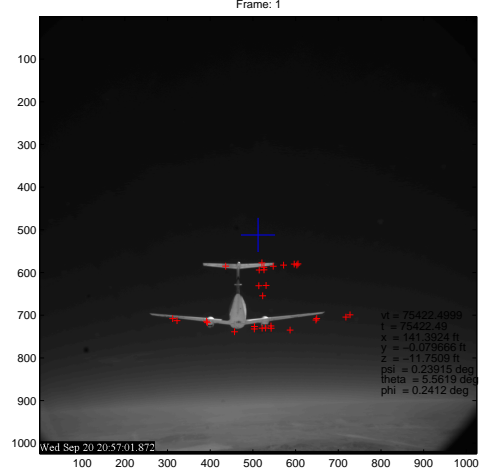
Figure 4.1:   Feature projection - (a) Feature projection based on the original TSPI data and camera parameters. (b) Feature projection using corrected TSPI data (compensated for angle of attack, wind drift, and sideslip) and the original camera parameters.

ified using a simulated annealing algorithm to optimize the values and reduce the error. The algorithm iteratively varied the camera parameters by a random amount and checked for a decrease in total error. If the error is decreased, the new parameters are kept. If the error was increased, the new parameters were kept with a probability that decreased through the iterations.

Two frames were chosen at different ranges from the C-12C, and ten features were used to calculate the projection error. The error metric was the sum squared error (SSE) of the radial distance between the projected feature location and the true feature in the image. The camera parameters were initialized based on the boresight measurements. The routine optimized the focal length and camera orientation, not the camera position. The camera position was included for one run of the algorithm; however, the run with only focal length and attitude led to a smaller total error. A normal zero-mean random variable and appropriate variance was added to each parameter during each iteration (e.g., 0.01 degrees for pan). The probability of accepting

an increase in error was

$$P_{accept} = e^{(-r*n*\delta_E)},$$

where δ_E is the change in error, r is a user-defined rate, and n is the iteration number. The parameters converged to a minimum SSE. In this case, the SSE reduced from 47000 to 1800 pixels^2, which is roughly a nine pixel error for each of the ten features. The resulting projection is shown in Figure 4.2.

(a)   (b)
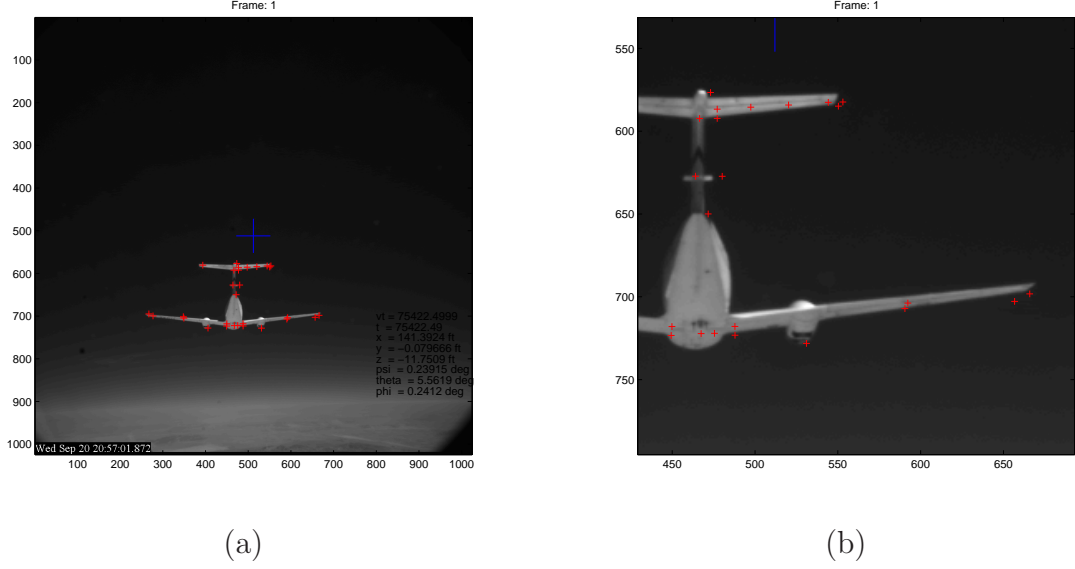
Figure 4.2:   Feature projection - (a) Feature projection based on corrected TSPI data and modified camera parameters. (b) Enlarged view of the feature projection.

The SSE could be optimized using a single frame, but the projection accuracy did not generalize to the entire data set as well. In addition to inaccurate camera parameters, the simple pinhole camera model could also contribute significantly to the error.

Although the projection quality was marginal, the resulting camera parameters were used for data analysis. It will be shown later that the projection was the dominant source of error in the vision system.

4.2 Baseline Refueling

The baseline refueling segment is taken from the second test flight, which occurred on 20 September, 2006. It consisted of a closure from the pre-contact to the contact position. It was representative of a typical refueling closure. The closure was made with an average closure rate of 1.4 feet per second. The closure was accomplished with a high sun angle ($> 45°$ from the horizon) on a clear day. One thousand four hundred frames were analyzed with an approximate frame rate of 30 frames per second. The truth data for each feature location in the image was hand-picked from every tenth frame and then interpolated to fill in each frame. The interpolation was done on the row and column data independently with a cubic spline interpolation function. (Note that this also had some variability.)

4.2.1 Image Point Estimation Performance. The error in feature projection, as discussed in Section 4.1 was the dominant source of error throughout the data analysis. For the baseline refueling, projection error increased as the receiver closed toward the contact position. Figure 4.3 shows the median error of each feature throughout the 1400 frame sequence. (For clarity, frame 1 corresponds to a range of 100 feet while frame 1400 corresponds to approximately 50 feet.)

The distribution of the error is shown in the histogram in the bottom left plot. It shows a bimodal distribution with modes at 2 and 8 pixels. After examining the histograms of each feature for the entire sequence, it was found that 18 of the 29 features used have bimodal distributions. Many of the projection errors increased significantly between the 600th and 850th frames (≈ 70 feet), which is reflected in the median plot. The reason some features exhibited a bimodal histogram while other features errors were relatively unchanged is unknown, however the error increased during momentary low-magnitude excursions in bank angle. Only the radial error is examined, although there is benefit to examining the direction of the error as well.
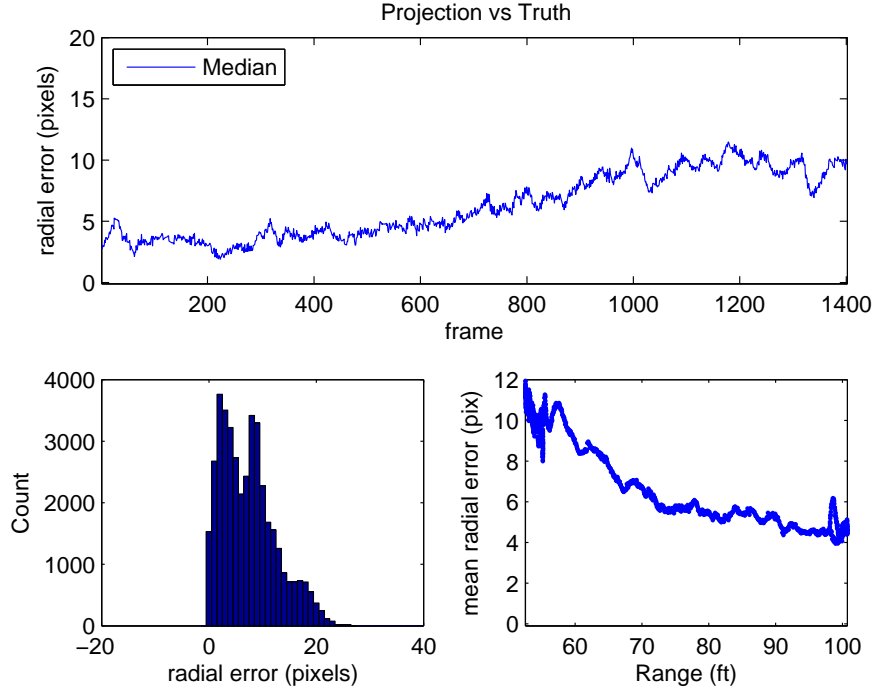
Figure 4.3:   Projection error - The radial error between the projected features and their true locations is shown. In the top graph, the median error is shown for each frame in the sequence. The lower left plot shows the histogram of the radial error over all 1400 frames. Since the video sequence is during a closure, the error is shown according to the range of the aircraft in the lower right plot.

4.2.2   Feature Detection.   The feature detection routine consisted of the modified Harris corner detector and the data association algorithm. For the following analysis, the observed position of a feature was the detected corner, which was associated with the feature-track by the association algorithms and was not necessarily the correct corner because of potential association errors. Some of these errors are discussed in Section 4.2.2.2

4.2.2.1   Feature Model.   Several features from the original feature model were discarded based on experience with the data. These points, as seen in Figure A.1, were points 6, 18, 20, 21, 25, 27, 28, 37, and 38. Points 6, 21, and 28 were discarded because they were rarely visible to the camera except at extreme azimuth and elevation angles. Point 38 was inadvertently excluded although it was

59

observed to be a strong feature. The other excluded points were discarded due to very low probability of detection, P_D.

The remaining points included strong and weak features. A strong feature was defined as a feature with a high probability of detection and low localization error (independent of the feature projection). A weak feature was any feature that was not a strong feature. For most features, the strength of the feature depended on range. Some features were weak at greater distances where the resolution of the camera was insufficient to breakout the corner. Examples of the metrics for feature strength are shown in Figure 4.4. These data were based on the baseline video segment and included ranges from 100 to 60 feet.
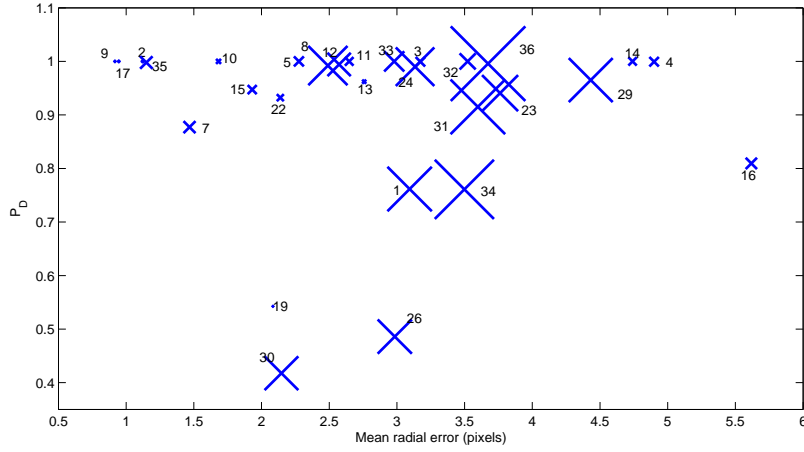
Figure 4.4:   Strength of Feature - The mean error of the feature detection algorithm is plotted verse the probability of detection. The number by each marker corresponds to the feature number in Figure A.1. The size of the marker is proportional to the variance of the error.

The strongest features (2, 9, and 17) were detected in every frame of the sequence with a mean error of less than 1.2 pixels and a variance of less than 0.4 pixels. The weaker features had poor localization (e.g. 31, 34, and 36) or a low P_D. These weaker features were more likely to contain association errors in the presence of projection errors or track propagation errors. Interestingly, paired features such as 4 and 11 did not share the same feature strength due to lighting during the sequence analyzed.

60

4.2.2.2 Data Association Issues. The problems with the image point estimation detailed earlier highlight the need for improved data association algorithms. Since the error associated with the camera projection is on the order of 12 pixels, the tracking gate must be large to allow the feature detector to find the correct corner. However, a large tracking gate increases the probability of association errors. The closest detected corner is associated with feature-track.

Figure 4.5 shows two examples of association issues caused by poor feature projection. In Figure 4.5(a), the lower left feature-track does not have a detected corner within its gate. The upper right feature-track is incorrectly associated with the lower left corner. In Figure 4.5(b), both feature-tracks are incorrectly updated by detected corners which are closer to the projection.

(a)   (b)
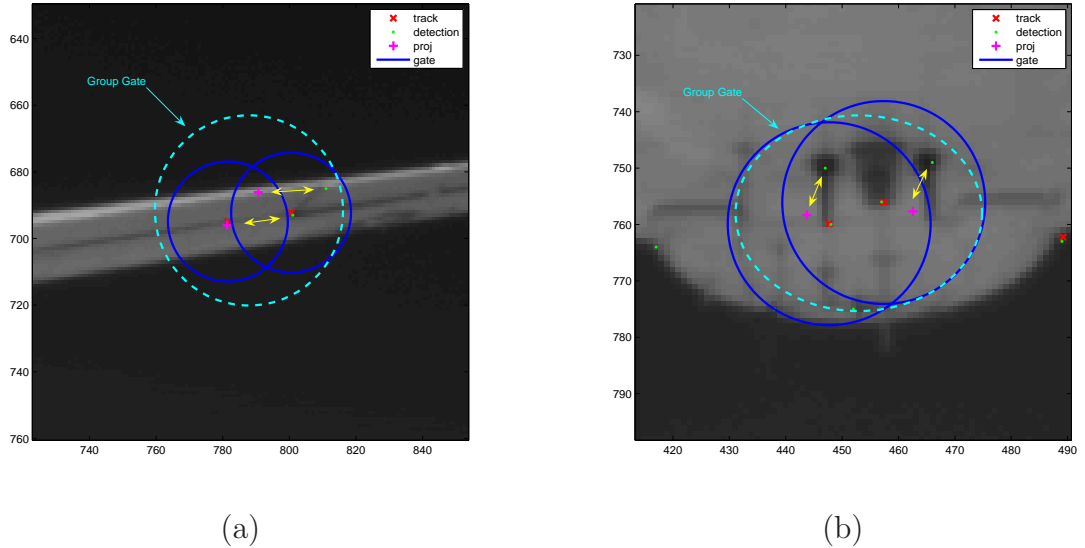
Figure 4.5: Association issues - Two examples of association error largely due to poor feature projection. A potential solution is shown by incorporating a group association scheme. (a) One feature is misassociated while another is starved of an observation. (b) Both features are misassociated.

A potential solution is to apply group tracking logic on features which are closely spaced. This logic should include the known structure of the group so that the features which most nearly match the structure of the group are associated with the individ-

ual measurements. Some group tracking methods are described by [2] and include techniques for individual target tracking supplemented by group information. The drawback to this method is the increased processing load. In this application, however, accurate association methods are necessary to provide accurate measurements to the navigation system.

The procedure could be expanded to include the entire feature structure of the tanker. It is not likely that the associations would improve overall, because the error in projection seems to vary by region of the aircraft. It should be noted, however, that the degree of complexity of data association logic depends on the accuracy of the feature projection. If the feature projection is accurate, a less complex data association algorithm is required.

4.2.2.3   Feature Detector Performance.   The following analysis compares the observed features in each frame of the sequence with the true (hand-picked) location of the features. The observed features are detected corners that have been associated to feature-tracks. The navigation updates $\hat{X}_b$ are provided every 30 frames, which is equivalent to approximately 1 Hz. This update rate is chosen to illustrate both the tracker performance and the effects of the updates. The predicted feature locations based on the navigation input and the camera projection contain the errors shown in Figure 4.3. Due to inaccurate projection, a constant radius gate of 13 pixels is used. Although the projection error is often greater than 13 pixels, this value is chosen to limit the number of misassociated tracks. The ZOH filter is also used as the baseline filter. The filter is important to the feature detector performance during frames in which a navigation update is not received because it determines the center of the gate for data association.

Figure 4.6 shows the median error over the length of the sequence as well as the histogram of errors and the errors by range. The sharp changes in the median of the radial error, such as at frames 930 and 960, coincide with navigation updates. The jumps that increase the error are largely due to feature-tracks being re-associated

62

with observations in closer to the projection and farther from the truth. The jumps that reduce the median error are primarily due to updated predictions which no longer have associated observations, and are therefore dropped. The radial error, on average, is improved over the error caused by the projection, which can be seen in Figure 4.3. This result is because the observed feature associated with a track is either correct or closer to the true location than the projection. In addition, features projected too far from the true location are starved of observations, and thus the track is dropped.

Figure 4.6:   Feature Detection Error (ZOH filter with projection error) - The radial error between the observed features and their true locations is shown based on the ZOH filter. In the top graph, the median error is shown for each frame in the sequence. The lower left plot shows the histogram of the radial error over all 1400 frames. Since the video sequence is during a closure, the error is shown according to the range of the aircraft in the lower right plot.

Comparison of the histograms in Figures 4.3 and 4.6 also shows reduction in error. The primary mode at 1 pixel of error contains 6378 hits, and the primary mode of the projection error in Figure 4.3 is at 2 pixels with only 3762 hits. The distribution became unimodal with the use of the feature detection algorithm. As

expected, using the gate of 13 pixels reduced the distribution above 13 pixels. The spike at -1 pixel error shows the number of unobserved features throughout the video sequence. The lower right plot illustrates the increase in accuracy over the projection error, particularly in the last 10 feet of closure (60-70 feet). This error is still greater than the desired accuracy of less than 3 pixels.

The same detection error is examined using the $\alpha - \beta$ filter (which changes the center of the tracking gates between updates), but the feature detection differences are insignificant.

The next question to answer is, "How would the feature detection performance change if the feature projection were near perfect?" The same set of frames is analyzed by updating the vision system with the 'truth' points every 30 frames instead of the feature projection. A gate of 6 pixels is applied instead of 13 because the larger gate is set to allow for projection error.

The error shown in Figure 4.7 is dramatically reduced. By improving the accuracy of the updates, a more realistic conclusion about the accuracy of the feature detection block can be drawn. The feature detection accuracy improves as the range to the tanker decreases due to improved resolution of the features. The association errors are also dramatically reduced due to both the accuracy of the updates and the smaller gates it allows. Note the increase the number of unobserved features as seen in the -1 column of the histogram, however, the improved accuracy of the detection scheme more than compensates.

Figure 4.7 shows the measurement accuracy attainable during accurate updates. This accuracy is of key importance, since the detected feature locations are the output of the vision system to the navigation system. In practice, the error seen by the projection could just as easily be caused by inaccurate pose estimates from the navigation system. In that case, the results in Figure 4.6 may be more realistic. The errors in the measurements can be characterized for use in the central-level tracking performed by the navigation system. In an ideal situation, the accuracy of the feature
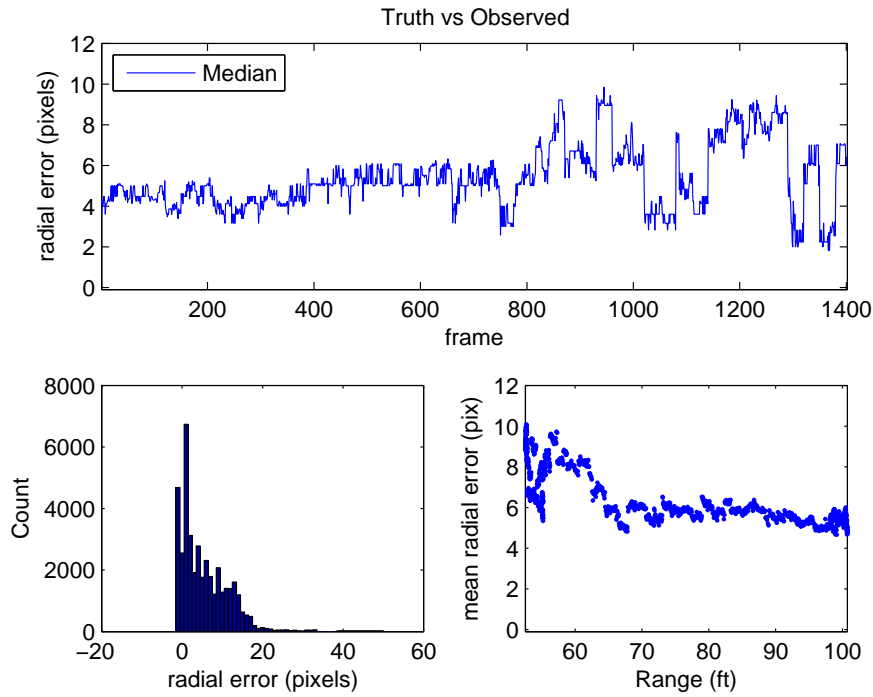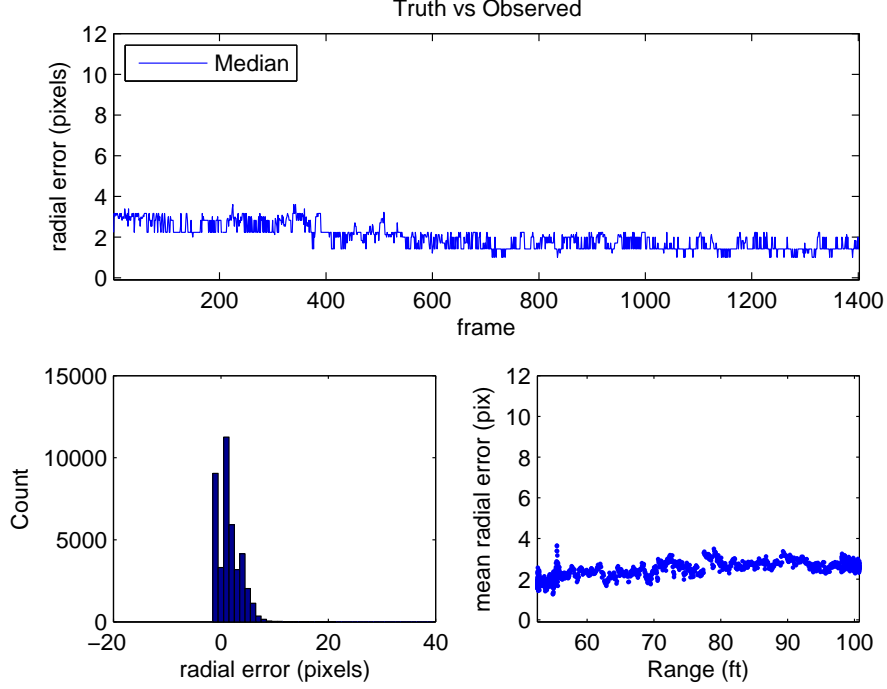
Figure 4.7:   Feature Detection Error (ZOH filter *without* projection error) - The radial error between the observed features and their true locations is shown based on the ZOH filter *using true feature locations for ≈1Hz updates.* In the top graph, the median error is shown for each frame in the sequence. The lower left plot shows the histogram of the radial error over all 1400 frames. The error is shown according to the range of the aircraft in the lower right plot.

detection converges with the accuracy of the navigation system to the levels seen in Figure 4.7. Another important conclusion is that with inaccurate feature projection, whether it is caused by the navigation update or the camera parameters, a reduction in navigation update rate improves the accuracy of the feature detection. This conclusion suggests that there is an update rate, depending on the projection error, that improves measurement accuracy.

4.2.3   Tracker Comparison.   The comparison of the $\alpha - \beta$ and the ZOH filter uses the same sequence of video as that in Section 4.2.2.3. The inaccurate feature projection is used, which drives the tracking gate to 13 pixels. The navigation update

rate remains every 30 frames. This time the estimated location of the features by the filters (the prediction) is compared with the true location in each frame.

4.2.3.1 Zero Order Hold Filter. The ZOH filter propagates the last observed location of the feature as the predicted location in the subsequent frame. It also drops the track after only one missed observation. The difference between the predicted location and the true location for the baseline refueling is shown in Figure 4.8. Since the projection error is dominant compared to the detection error, the filter error is driven by the misassociations. The sharp changes in the top plot correspond to the navigation update.

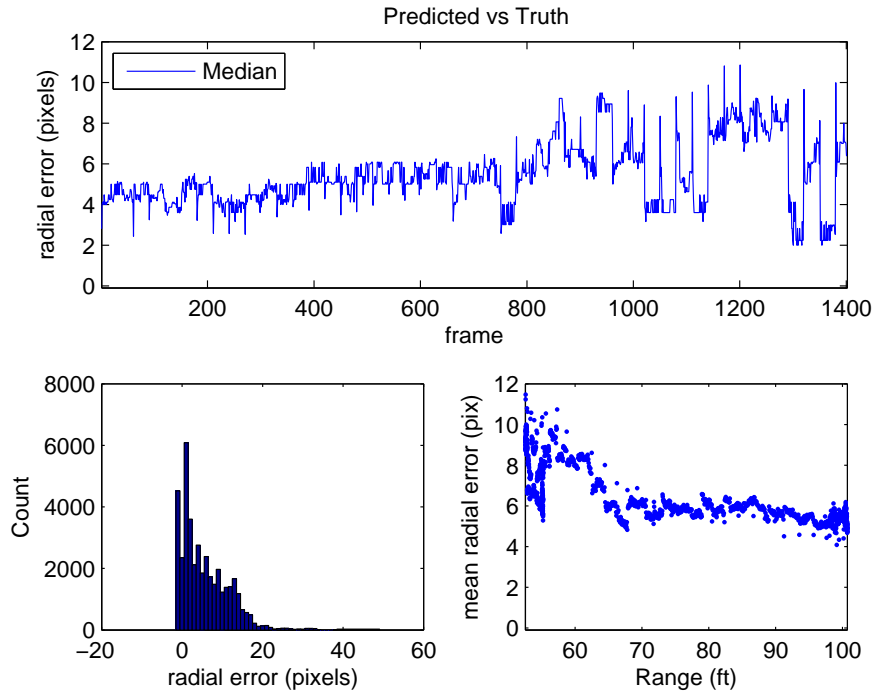Figure 4.8: Filter Performance (ZOH filter with projection error) - The radial error between the predicted feature location and their true locations is shown based on the ZOH filter. In the top graph, the median error is shown for each frame in the sequence. The lower left plot shows the histogram of the radial error over all 1400 frames. The error is shown according to the range of the aircraft in the lower right plot.

Figure 4.9 shows the number of unobserved features and dropped tracks for the baseline sequence using the ZOH filter. The spikes correspond to a navigation update where all feature-tracks that should be visible in the image are initiated. For the entire sequence, the average filter error (predicted location minus true feature location) is 6.56 pixels.
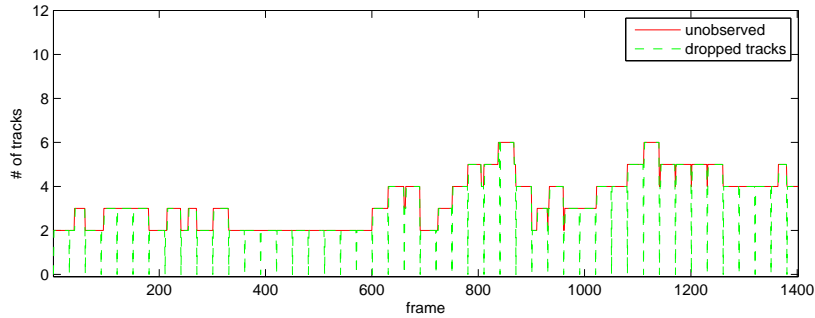

Figure 4.9: Dropped tracks (ZOH filter) - The number of unobserved features and dropped tracks in each frame are shown for the baseline sequence using a ZOH filter.

4.2.3.2 $\alpha - \beta$ Filter. The $\alpha - \beta$ filter, unlike the ZOH filter, propagates the estimated location of the feature-track based on feature estimated velocity. It also retains tracks for features that are not observed until the integer track score reaches the deletion threshold. For those unobserved tracks, the estimated position is propagated based on the last velocity estimate. For this run, the position is updated with a gain of $\alpha = 0.7$ and a velocity gain of $\beta = 0.075$. These values are determined empirically, because they generalize better to other less benign sequences. It also makes the estimate more robust when dealing with spurious measurements or 'false' corners.

Figure 4.10 shows the errors associated with the $\alpha - \beta$ filter. The difference between the $\alpha - \beta$ filter and the ZOH filter are very small in terms of error. The average error for the $\alpha - \beta$ filter is 6.84 pixels. There is a slight increase in error because the position update lags the movement of the detected corner.
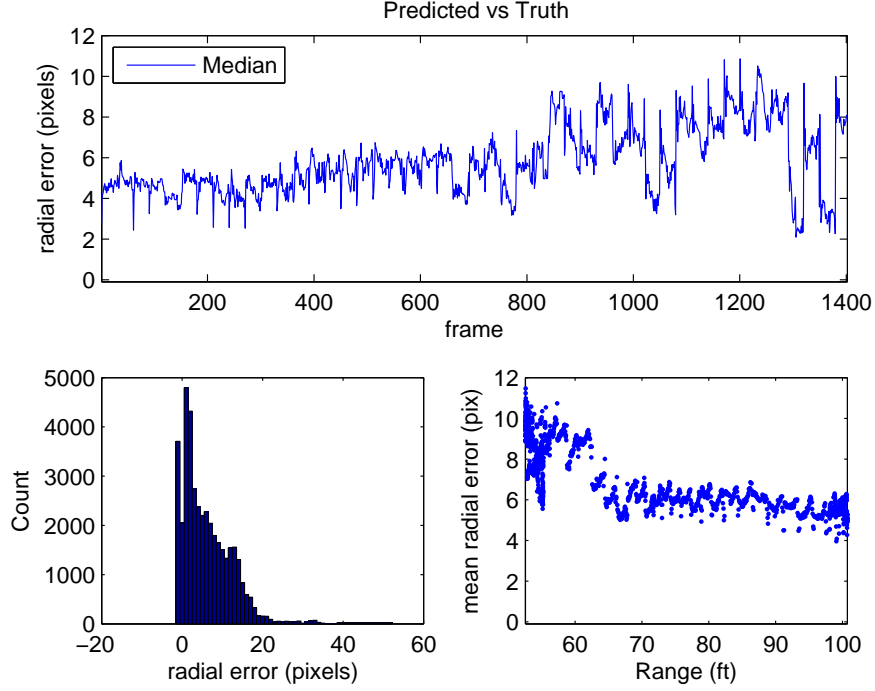
Figure 4.10: Filter Performance ($\alpha - \beta$ filter with projection error) - The radial error between the predicted feature location and their true locations is shown based on the $\alpha - \beta$ filter. In the top graph, the median error is shown for each frame in the sequence. The lower left plot shows the histogram of the radial error over all 1400 frames. The error is shown according to the range of the aircraft in the lower right plot.

In addition to being more robust to spurious corners, the $\alpha - \beta$ filter provides more observations than the ZOH filter, because it does not drop tracks based on a single missed observation. The use of a track score for track deletion allows the detector to reacquire features after missed detections. This fact is evident by examining Figure 4.11. In all cases, the $\alpha - \beta$ filter has the same or fewer dropped tracks. The unobserved features are later acquired.

Although the difference in the filters is minimal for this benign case, there is one advantage to the $\alpha - \beta$ filter: for very little cost in accuracy, additional observations are gained. In the next section, the ZOH filter is again compared to the $\alpha - \beta$ filter for a more dynamic video sequence.
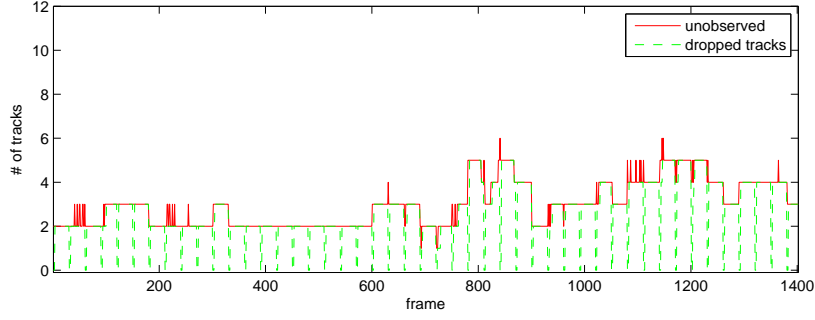
Figure 4.11:   Dropped tracks ($\alpha - \beta$ filter) - The number of unobserved features and dropped tracks in each frame are shown for the baseline sequence using a $\alpha - \beta$ filter.

4.3 High Rate Closures

The high-rate refueling segment is taken from the same flight on 20 September, 2006. The closure was accomplished with a high sun angle ($> 45°$ from the horizon) on a clear day. One thousand frames were analyzed with an approximate frame rate of 30 frames per second. The truth data for each feature location in the image was hand-picked from every tenth frame and then interpolated to fill in each frame. The interpolation was done on the row and column data independently with a cubic spline interpolation function.

The video segment begins with a quick closure from the pre-contact to the contact position followed by a backing-out segment, a normal closure to contact, and finally a quick lateral movement to the right. The initial closure occurs from frames 1-415 at a rate of 3.6 feet per second. The LJ-24 then backs out 15 feet from frames 415-570 at a rate of 2.8 feet per second. Frames 570-765 consist of a straight closure at 1 feet per second. The final segment from frames 765-1000 consist of a lateral movement to the right at 1.1 feet per second. The initial closure and the lateral movement are greater than normal rates for refueling.

The movements of both aircraft directly affect the movement of the feature-targets in the images. This movement must be considered when choosing the size of the gates used during data association. As stated earlier, the motion of the receiver

69

is primarily along the optical axis of the camera. Objects in the camera typically move radially from a focus of expansion (FOE) during closure. Feature movement is inversely proportional to the range from the target. In addition, features accelerate as they move toward the edge of the image. During the lateral movement toward the end of the segment, the points move together in one direction on the image. After analyzing the movement of each feature, the maximum displacement of a feature in one frame is four pixels. The average movement is only 0.66 pixels per frame (ppf) during the high-rate closure, 0.46 ppf during the normal closure segments, and 1.22 ppf during the lateral segment.

The errors caused by the projection of the feature model are consistent with those shown in Section 4.2.1. To create a better comparison, the navigation updates are not used. Instead, the true feature positions are used to update the vision system every 30 frames. The feature detection errors with true updates are consistent with those in Figure 4.7. The reduced update errors allow for smaller gate size, which in turn also reduces the association errors.

Based on a maximum feature movement of 4 ppf and an average feature detection error of 2-3 pixels, a gate of 9 pixels is chosen for the following analysis. With a smaller gate and more accurate updates, the performance of the trackers can be better isolated from projection errors and data association issues.

4.3.1 Zero Order Hold Filter. The ZOH filter is able to maintain the tracks of more than two thirds of the features through the high-rate video segment. The average tracking error is 3.14 pixels between the predicted location of the features and their true locations in the image. Figure 4.12 shows the median filter error between two and four pixels throughout the sequence. There are 4573 missed observations due to dropped tracks throughout the sequence. The majority of the predictions are less than four pixels from the true feature location.

As the receiver approaches contact, the error in the predictions improves slightly. It is also shown that the updates create a brief zero-error. Following detection, the

70

prediction moves to the detected corner location. The errors in the prediction are more consistent with the feature detection errors.
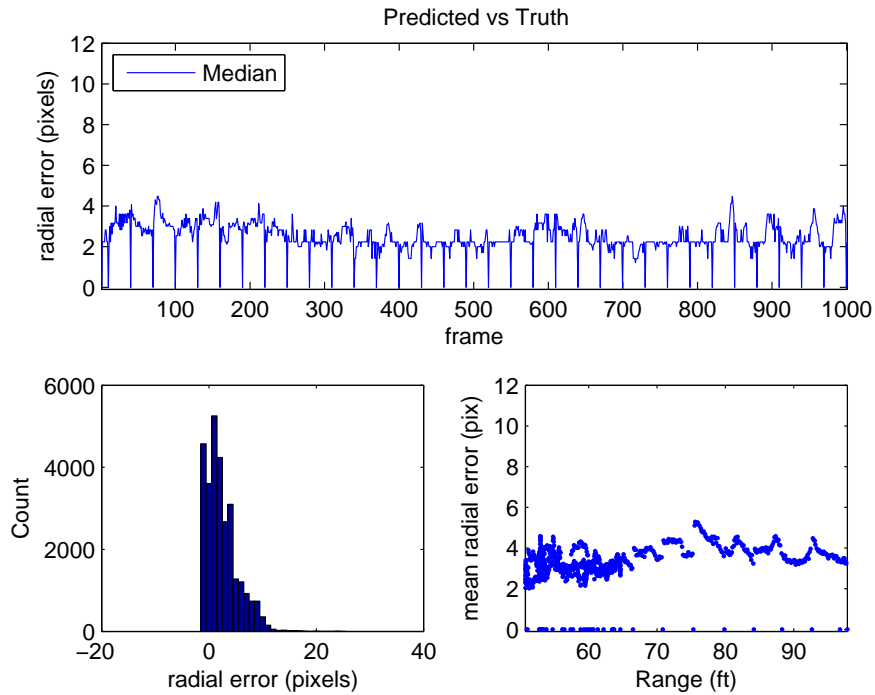

Figure 4.12: Feature Detection Error (ZOH filter *without* projection error) - The radial error between the predicted feature location and their true locations is shown based on the ZOH filter. The true feature locations are used as the navigation update and a smaller association gate of 9 pixels is used.

4.3.2 $\alpha - \beta$ Filter. The $\alpha - \beta$ filter brings an added benefit of using a track score for determining track deletion rather than the first missed observation. For this simulation the position gain (α) is 0.9 and the velocity gain (β) is 0.25. These numbers were found empirically based on methodical variation of the position and velocity gains.

Figure 4.13 shows the median error over time and the mean error with distance, which were very similar to the ZOH results. The average filter error increases slightly over the ZOH to 3.35 pixels per feature. The notable change is that there are only 3068 missed observations due to dropped tracks, which means that there is an average

of 1.5 more observations per frame. This is primarily due to the addition of a track score for track deletion.
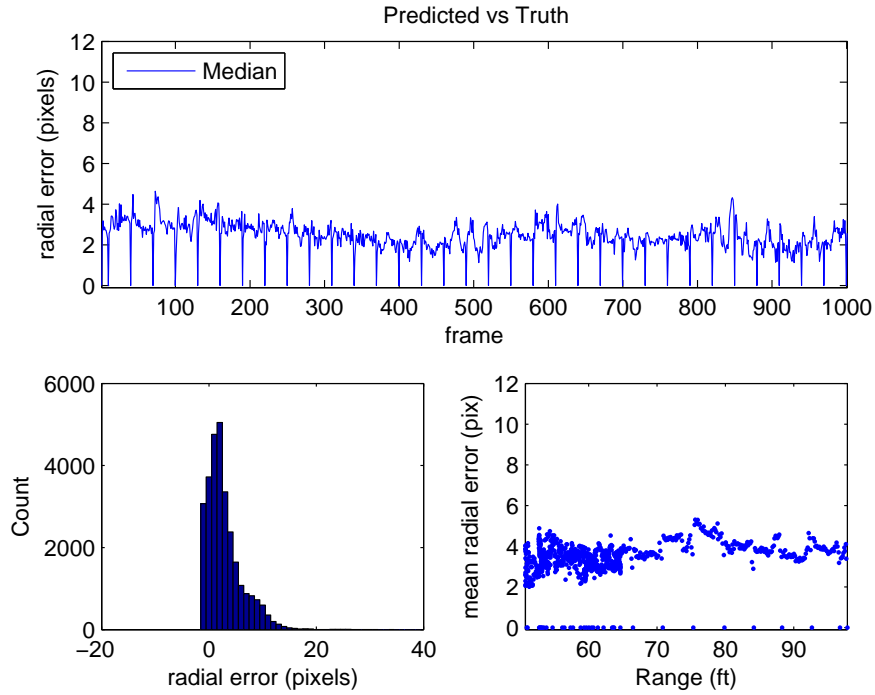

Figure 4.13: Feature Detection Error ($\alpha - \beta$ filter *without* projection error) - The radial error between the predicted feature location and their true locations is shown based on the $\alpha - \beta$ filter. The true feature locations were used as the navigation update and a smaller association gate of 9 pixels was used.

The ability of the $\alpha - \beta$ filter to maintain weaker tracks can be seen in Figure 4.14. It also maintains some tracks which are lost by the ZOH filter because of their velocity. One reason that the number of dropped tracks is reduced during the lateral movement segment (frames 765-1000) is that as features exit the FOV, they are no longer considered 'dropped' tracks. The term "dropped" tracks refers only to tracks that should be visible in the image.

Comparison of the ZOH and $\alpha - \beta$ filters shows that both filters provide approximately the same error in predicting the location of the true feature. This filter error is roughly equivalent to the feature detection error (with accurate track initiation and updates). Throughout the sequence, there are a minimum of 20 observations
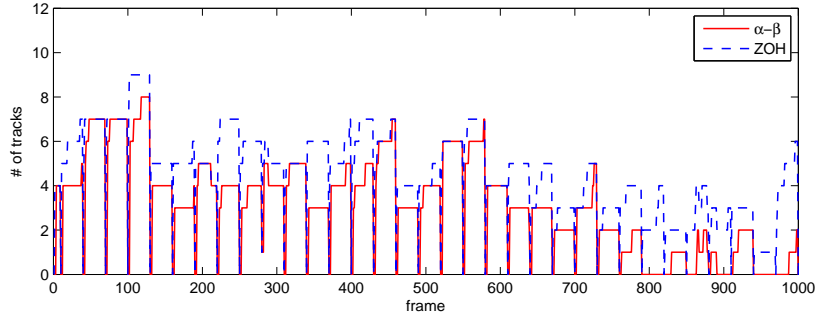
Figure 4.14: Dropped Track Comparison - The number of dropped tracks for the ZOH and $\alpha - \beta$ filters are shown for the high-rate video segment.

per frame which is more than is necessary to calculate a pose estimate of the tanker aircraft.

The $\alpha - \beta$ filter is able to provide an average of 1.5 more observations per frame than the ZOH filter which came at a very slight increase in error and computational expense. In both cases, the computation expense is minimal when considering the entire design.

Figure 4.15: Filter Performance Comparison - With a track score, the ZOH filter has fewer dropped tracks and an increased error. The α-β filter has improved accuracy over the ZOH filter with a track score.

Overall, the α-β filter increases the number of measurements with a slight increase in error. The ZOH, when modified to include a track score for track maintenance and deletion, has more observations and a slightly increased filter error. Figure 4.15 shows the comparison with exaggerated differences in error.
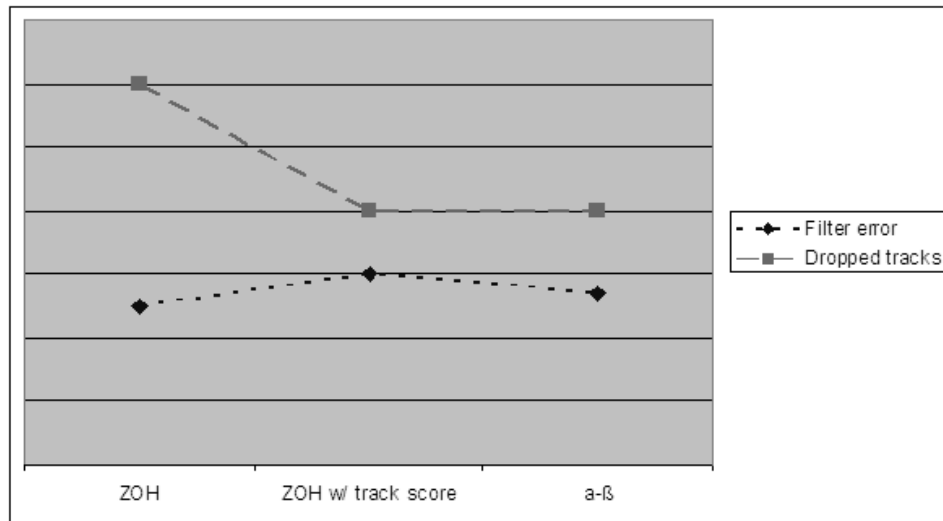
4.4 Environmental Factors

Several environmental factors can limit the effectiveness of the feature detection. These environmental factors increase the image noise. Some of these factors include clouds in the background of the scene, low light situations (such as near sunset), and the extreme low-light case, night. Although refueling can and does occur in clouds, these conditions were not flown during the flight test. The following analysis is done at a more qualitative level.

4.4.1 Cloudy Background. Clouds in the background creates two problems for the feature detection. First, the number of false corners in the background is greatly increased due to gray level gradients in the clouds. Second, there is an increase in the saturation of pixels in the images. Combined, these two effects cause false associations and diverged tracks, especially when the projection of features is inaccurate.

Figure 4.18 shows a sample image taken from the fourth flight. The image is taken at a range of 96 feet from the C-12. The image illustrates the saturation of pixels, particularly on the right side of the aircraft. The clouds in the background also create corners in the image which are detected by the corner detector.

Two hundred frames of the sequence were analyzed with and without the projection errors included. The projection error in this sequence had a mean of 12.1 pixels from the projected location of the features to the true feature locations. Both trials used a ZOH filter. To account for the projection error and feature detection error, an association gate of 16 pixels was used.
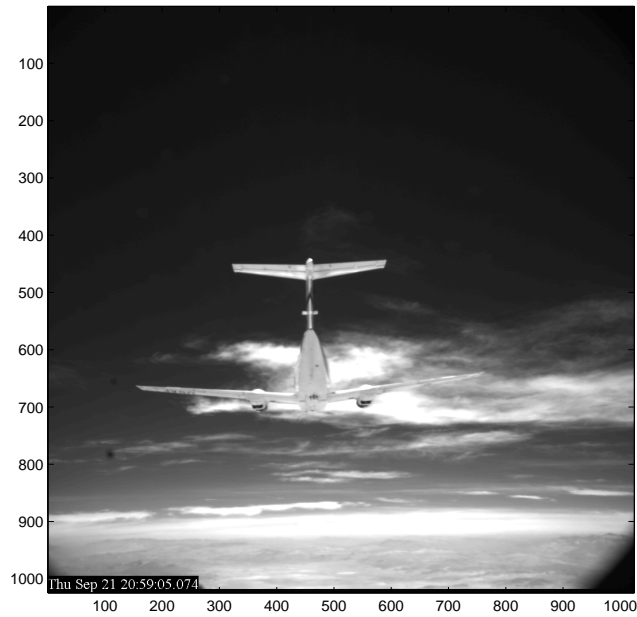
74

Figure 4.16: An image is shown which contains clouds in the background of the scene. The presence of clouds and background clutter degrades the feature detection and tracking algorithms.

The first trial of the video segment used the true feature locations to negate the projection error effects. Although a significant portion of the C-12 was saturated in the sequence, there were very few dropped tracks. A dropped track is preferable to a divergent track, i.e. no measurement is better than a bad measurement. The image saturation caused several misassociations as well as increased error in localization of the features. The feature detection error had a mean of 5.1 pixels. By comparison, the baseline closure (which was free of background clutter) had a mean error of 2.7 pixels at a range of 90 feet. Even with the localization errors, very few of these tracks diverged.

The second trial used the projected features as the updates, which introduced significant error in initiating tracks. With the initial tracks in error by an average of 12 pixels, there were a large number of misassociations, an increased number of dropped

75

tracks, and several diverged tracks. There were also several tracks with a relatively low feature detection error. In terms of feature strength, features 2,5,9,14,24,31,35 were relatively strong.

Figure 4.17 shows histograms of the feature detection errors for both the first and second trial. The localization errors are apparent by the mode shift to four pixels. Based on experience, strong features should always be detected within six pixels of the true location (although this number varies with range). The tracks with errors of ten or more pixels indicate misassociations, and the right-side tail indicates some diverged tracks.
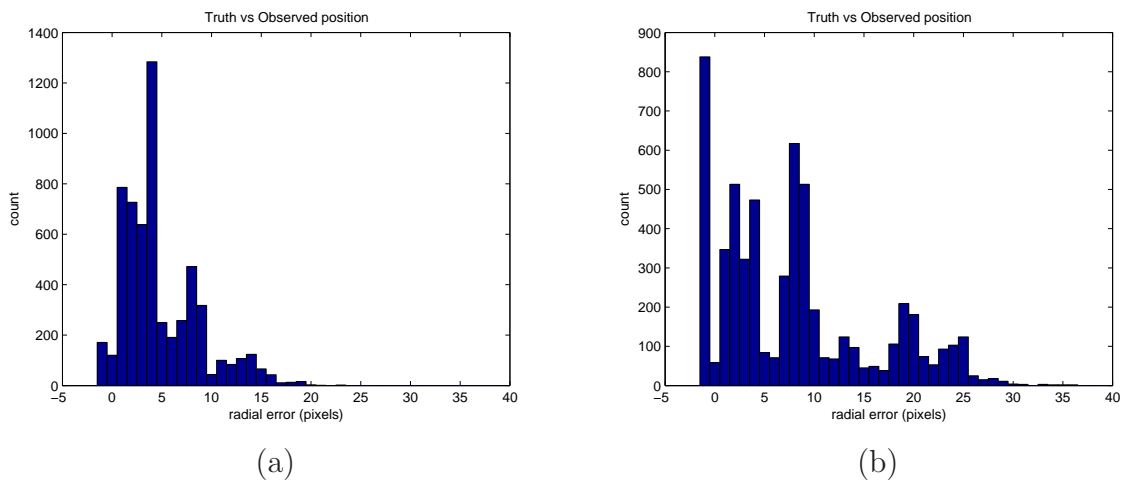
Figure 4.17:   Feature Detection Comparison - The histograms of the feature detection error are shown for a video segment with clouds in the background. (a) Using the true feature locations to initiate and update tracks (b) Using the navigation update with feature projection.

The histogram for the second trial indicates an increased number of unobserved tracks (-1 column) due to the projection error. The increase in the stretching of the histogram is the largest indicator of diverging tracks, especially when the spread exceeds the gate size. Figure 4.18 illustrates two diverging tracks caused by the clouds, and one poorly localized track caused by pixel saturation on the right side of the fuselage.

76
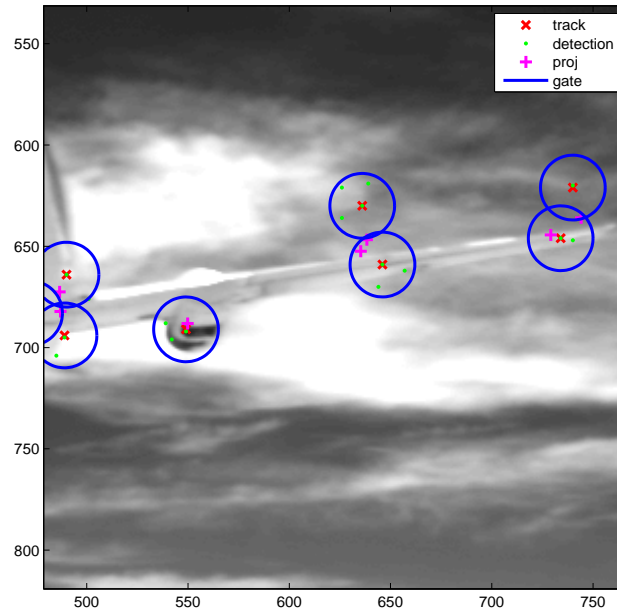
Figure 4.18:   Divergent Tracks - An enlarged portion of
Figure 4.18 is shown with the projected feature locations,
detected features, tracks, and association gates. This
portion illustrates two diverged tracks which are tracking
points in the clouds.

The presence of clouds significantly reduces the accuracy of the feature detection
and increases the probability of diverged tracks. The probability of diverged tracks
increases when the accuracy of the track initiation and updates is degraded. Pixel
saturation indicates the need for automatic f-stop adjustment, a greater dynamic
range in the camera, or both.

4.4.2   Low Sun Angle and Low Light.   The low sun angle causes major issues
when the sun is within the camera FOV. For the camera and filter used for this test,
most of the images were saturated near the horizon when the sun was within the
camera FOV, as can be seen in Figure 4.19. The saturation made feature extraction
nearly impossible while the sun was within the FOV.

Another factor is the dynamic range of the camera versus the dynamic range of the scene. The pixel elements in the CCD array collect photons and quantize their number. If a pixel collects more photons than its maximum value, the value is clipped and detail is lost. This overflow can also cause a blooming effect, in which this charge spills into surrounding pixels. The areas on the horizon can be caused by blooming or atmospheric refraction detected by the camera but outside the visible spectrum.

Figure 4.19:   Three frames with the sun in the FOV.

Although some of the features that lie on the silhouette could be extracted, features within the body of the aircraft are indistinguishable. The camera configuration in this case is fixed, with a manually adjustable f-stop, which determines the input pupil size. Without adding significant optical filters, automatic f-stop features, and potentially costly image processing, the vision system is ineffective in this situation. Without these improvements, this vision system can not be used with the sun within the FOV, which limits air refueling operations significantly, although pilots experience the same safety concerns while refueling into the sun. The standard solution is to change the track by several degrees until the refueling can safely proceed, which should not be a factor for AAR.

The images in Figure 4.20 are taken during the same period with the sun outside the FOV. In the first two cases, the direct light on one side of the aircraft illuminates the details, which allows for decent feature extraction. The side opposite the sun is shadowed and the details are no longer visible. Although the details are not visible, the corner detector is still able to extract many of the corners at close ranges. There are also a greater number of false corners detected in the shadowed areas. The corner

detector has much more trouble with low light situations when there are clouds in the background.

Figure 4.20:   Three frames with the sun outside the FOV.

During twilight, the features on the silhouette are both visible and detectable by the feature extraction algorithms. This detection can be aided by enhancing the contrast of the image through histogram equalization techniques. Histogram equalization also tends to amplify the noise in the image, causing several false corners. These false corners can lead to degraded and divergent tracks.

Figure 4.21:   Three frames taken during twilight.

4.4.3 Night.   The camera used in this test is not designed for night use. The images taken after twilight are essentially black with no useful information. Normal image enhancement techniques are unable to modify the image enough to extract any useful features, which highlights a significant limitation of using a sensor that is sensitive to only the visible spectrum. Since modern warfare is a 24-hour, all-weather event, an EO camera is an insufficient sensor for AAR at night and in weather.

79

4.5 Lens effects

For the final flight of the test, a lens with a focal length of 25mm replaced the 12.5mm lens used on the previous five flights. Changing the focal length effectively did two things. It provided better resolution of the scene and limited the FOV. The new field of view was approximately 27 degrees versus the 52 degree FOV for the 12.5mm lens. Although the FOV changed, the pixel coverage remained the same. It also effectively created a 2x optical zoom.

The segment analyzed contained 200 frames of a closure from 97 to 80 feet. The closure was made with a high sun angle, and there were no clouds in the background. The true feature locations were used to initiate and update tracks every 30 frames. A ZOH filter with a gate of 6 was chosen for comparison with Figure 4.7. The results of the 200 frame trial are shown in Figure 4.22.
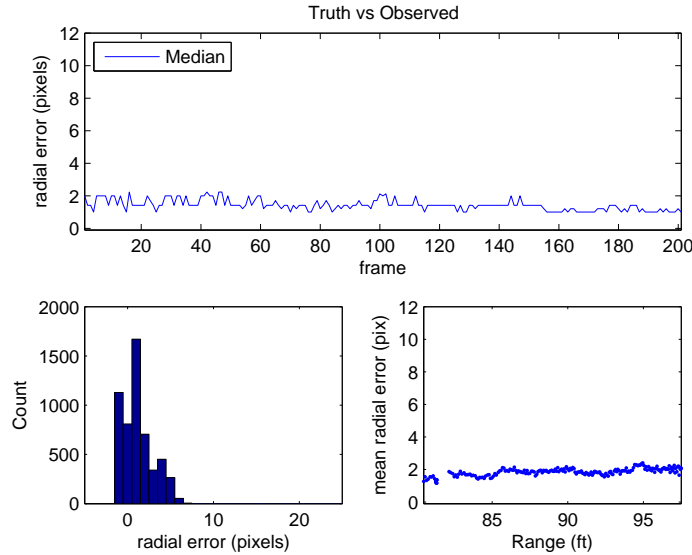

Figure 4.22:   Filter Performance (ZOH filter *without* projection error) - The radial error between the predicted feature location and their true locations is shown for the video segment using the 25mm lens. The results are based on the ZOH filter with a gate of 6 pixels. The true feature locations were used to initiate the tracks and update them every 30 frames.

The mean feature detection error was 1.96 pixels, which was a 0.75 pixel improvement. The improved feature detection was attributed to the increase in resolution given by the longer lens. The drawback to the longer lens was the restricted FOV. With the restricted field of view, fewer features were visible. For instance, in Figure 4.23, four features have exited the FOV prior to a range of 80 feet. In contact, neither the horizontal stabilizer or the wings outboard of the engine nacelles were visible.
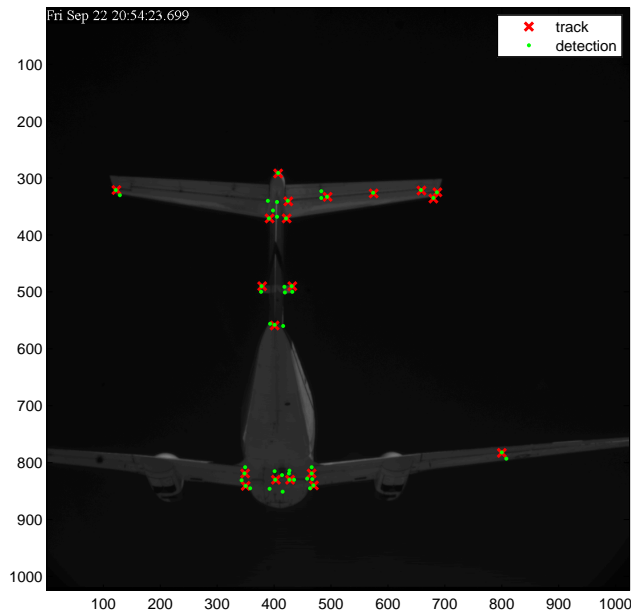
Figure 4.23: A sample frame from the 25mm lens is shown with the tracks and detections. The aircraft range is 80 feet, and four features from the feature model are already outside the FOV.

The 12.5mm lens, combined with a larger tanker such as a KC-135, would exhibit the same characteristics. Because of the size of the KC-135, one would expect there to be many more detectable features on the fuselage and the inner one third of the wings. For application, it would be better to have a wide angle lens or multiple narrow angle lenses.

4.6 Algorithm Speed

The vision system in this thesis was implemented in MATLAB® and the simulations were run on a desktop computer with a 2.19Ghz processor and 1.0 GB of RAM. The video, due to its size, was accessed via an ethernet crossover cable connection from a Buffalo TeraStation. The Buffalo TeraStation contained four SATA drives for a total of 1.8 TB and a link speed 1000Mbps. Each image contained approximately 2MB of data.

The total run-time of the vision system was approximately 0.4 seconds per frame including overhead. The majority of the time, 0.22 seconds per frame, was spent reading the images into MATLAB®. The second most costly subroutine was the Harris corner detector, which took about 0.09 seconds per frame. The MATLAB® code included several extraneous lines of code dedicated to error checking, displaying output, and saving simulation data which occupied up to 5% of the time per frame.

The actual application of this vision system is realizable in real-time if two conditions are met. First, it would have to be optimized and hand-coded into a high level programming language such as C++, which should provide a ten-fold increase in speed. At that rate, this algorithm would run at 55 frames per second (not including image loading). Second, the image retrieving method would have to be faster than 0.015 seconds per frame.

V. Conclusions and Recommendations

Future UAVs will require an air refueling capability to increase their range and endurance. Since air refueling forces the close proximity of aircraft, there is no room for miscalculation in the navigation system. To achieve an automated air refueling capability, the Air Force Research Laboratory has been seeking to develop a combination of GPS, inertial, and vision sensors to achieve the accuracy and reliability necessary for successful automated aerial refueling operations.

The vision sensor brings a passive sensor that can operate with no tanker modifications. The challenge in using a vision sensor for AAR is estimating the relative position of the tanker aircraft from an electro-optic (EO) sensor.

The method investigated in this thesis involves identifying points of interest in the video of the tanker and calculating three-dimensional vectors to these points in the camera frame. These vectors can be passed to a navigation integration system for the final relative position determination. The system design is tightly coupled with the navigation system in that it does not compute an optical-based position and attitude solution prior to integration with the inertial measurements. The navigation system, which is not the subject of this thesis, can use the feature measurements directly.

5.1   Conclusions

The vision system described in this thesis is a viable solution to relative navigation for AAR with a few caveats. The algorithm works in simulation using real world video and TSPI data. The system is able to provide at least a dozen useful measurements per frame, with and without projection error. The vision system used here is far from the matured state required for operational use. The feature projection to initiate and update tracks needs significant improvement. There are also ways to improve the feature extraction and tracking functions. In addition, the EO sensor used in this test limits the vision system to daylight conditions in good weather.

The estimation of the features on the tanker in the image is the dominant source of error in the design. This error is caused by the combination of the navigation system input to the vision system and the camera model. The navigation input contains the relative position and orientation of the tanker aircraft which is used to locate the tanker features of interest. The tanker model is then projected using a pinhole camera model. The camera model uses the camera position, orientation, and focal length for the projection. With the combined inaccuracies of the navigation system and the camera model, the projection of the features onto the image is marginal with an average difference of 13 pixels from the actual feature location in the image. Although the projection quality is marginal, the resulting camera parameters are used for data analysis in several simulations. In these simulations feature projection is the dominant source of error. Feature projection is extremely important because it is the basis for initiating and updating feature-tracks. The vision system design here is heavily dependent on the accuracy of the navigation updates. It is not yet robust enough to handle situations where the navigation update is considerably inaccurate.

The feature detection block consists of the modified Harris corner detector and the data association algorithm. The detection and association accuracy of features depends on the strength of the feature, which is defined as a feature with a high probability of detection and low localization error. The strength of a feature changes with respect to the range of the aircraft, the environment, and the quality of the images. Weaker features are more likely to contain track association errors. The basic strength is based on the best environmental conditions and good image quality (lighting, contrast, focus, etc.). Based on basic feature strength, the feature model should be modified to add previously unmeasured strong features and remove weaker features.

In this application, accurate association methods are necessary to provide accurate measurements to the navigation system. Several data association issues appeared in the presence of weak features and poor projection quality. In these cases, some

tracks are starved of observations, while others are misassociated. These association errors lead to feature detection errors.

The mean feature detection error is 2.7 pixels using the 12.5mm lens and 1.95 pixels for the 25mm lens with a clear background. This level of accuracy should be very useful to the navigation system in determining the relative position of the tanker aircraft.

The feature detection error increases significantly for cloudy backgrounds due to image saturation and misassociations. During low sun angles while the sun is within the camera FOV, vision system can not be used (with the test configuration). Without adding significant optical filters, automatic f-stop features, and potentially costly image processing, the vision system is ineffective in this situation. While the sun is outside the FOV, the details are not visible, but the corner detector is still able to extract many of the corners at close ranges. There are also a much greater number of false corners in the shadowed areas. During twilight, only silhouette features are reliably detected. Feature detection using the tested EO camera is not possible during night conditions.

The use of a lens with a longer focal length decreases the FOV and increases the resolution of the tanker, which provides better feature detection accuracy but significantly decreases the number of measurements available. Since the increase in accuracy is small, it would be better to have a wide angle lens or multiple narrow angle lenses.

Two sensor level tracking methods are evaluated using the baseline closure and high rate movements by the receiver aircraft. The ZOH filter is the simplest and propagates only the last location of the feature-tracks to the next frame. Tracks that are unobserved are dropped after a single missed observation. The $\alpha - \beta$ filter propagates the location of the feature-tracks to the next frame based on the estimated location in the current frame and the estimated velocity of the track. It also uses an integer track score and threshold for track deletion.

The $\alpha - \beta$ filter exhibits a slight increase in error over the ZOH filter because of position update lag and track maintenance on weaker features. The lag is designed into the filter to reduce the effects of misassociations to false features. The $\alpha - \beta$ filter provides more observations than the ZOH filter due to the use of the track score for track deletion. Because it does not drop tracks based on a single missed observation, it allows the detector to reacquire features after missed detections.

The comparison of the ZOH and $\alpha - \beta$ filters shows that both filters provide approximately the same error in predicting the location of the true feature. This filter error is roughly equivalent to the feature detection error (with accurate track initiation and updates). Although the difference in accuracy of the filters is minimal, there is one advantage to the $\alpha - \beta$ filter: for very little cost in accuracy, additional observations are available. The $\alpha - \beta$ filter is able to provide an average of 1.5 more observations per frame than the ZOH filter, which comes at a very slight increase in error and computational expense.

The bottom line is that while both filters were simple, they both have satisfactory performance when used with accurate track initiation and updates. The ZOH filter works well in low dynamic situations, and the $\alpha - \beta$ filters improves robustness and track maintenance of fast moving tracks. There appears to be no need for a more complex tracker at the sensor level.

The blending of sensor level tracking and navigation updates is not evaluated in detail. In fact, the 'blending' term is a misnomer for this application. Development of a blending algorithm is necessary to ensure continuous accurate measurements of features. The projection errors decreases measurement accuracy and leads to misassociated and dropped tracks, which causes accurate tracks to be dropped due to an inaccurate update. The benefit of the track updates is apparent during decent feature projection. Tracks that have drifted or been misassociated are dropped and the correct tracks are initiated. The complexity of the blending function depends on the accuracy of the navigation updates and feature projection. If the feature projection

is poor, it must be compensated by less frequent updates or by adding more complex blending logic such as MHT.

The speed of the vision indicates that this design is easily feasible in real-time. The total run-time of the vision system is approximately 0.4 seconds per frame, including overhead. The majority of the time, 0.22 seconds per frame, is spend reading the images. The second most costly subroutine is the Harris corner detector, which takes less than 0.09 seconds per frame.

The actual application of this vision system is realizable in real-time if two conditions are met. First, it would have to be optimized and hand-coded into a high level programming language such as C++, which should provide a ten-fold increase in speed. At that rate, this algorithm would run at 55 frames per second (not including image loading). Second, the image retrieving method would have to be faster than 0.015 seconds per frame.

The operational utility of this vision system is marginal due to its degraded performance during less than ideal environmental conditions. The use of an EO camera does not provide the all-weather capability required for AAR. The vision system design should be easily transferable to a more robust sensor, such as an infrared sensor or fused EO/IR system.

5.2 Recommendations for Future Research

Although feature detection and tracking performance appear satisfactory, a judgement about the combined vision and navigation system cannot be made. Before seeking these improvements, the whole design including the navigation system should be tested.

Image point estimation needs improvement with better camera calibration for the camera parameters. A more complex and accurate camera model may be required to achieve the desired accuracy. It is recommended that future work use a more sophisticated camera model and methods to obtain the camera parameters.

87

Feature extraction could be improved by improving the detection of weaker features and data association. Feature unique masks could be added to aid in detecting weaker features. If many strong features available, this addition would be unnecessary. Weaker features could be eliminated from the model or de-weighted in the navigation system.

A potential solution to some of the data association issues is to apply group tracking logic on features which are closely spaced. This logic should include the known structure of the group so that the features which most nearly match the structure of the group are associated with the individual measurements. Dynamic gating could also be used based on a covariance matrix or on Mahalanobis distance. Gating could also be changed based on the track score or feature strength.

The blending of pose-based feature estimates and the sensor level tracking estimates should be improved. One possible method is the use of multiple hypothesis testing (MHT) on features that are not close to each other. In this case both pose estimates and tracked features would be evaluated for a few cycles to determine the best estimates.

To improve tracking, the global structural motion could be incorporated to discard tracks that are moving in the wrong direction, although process could be accomplished by the navigation system as well. In addition, by implementing the idea of the focus of expansion, simpler filters could be used when movement is benign, and more advanced filters could be used when movement is more dynamic.

Another area for research is the addition of an acquisition and alignment function for the vision system. The acquisition of the tanker and feature points based only on the vision system is a considerable task. It would however reduce the dependence on the navigation pose estimate. To augment the acquisition, an alignment algorithm could be added to correlate the projected features and the globally detected corners, which could help reduce the errors in the camera parameters and navigation input.

Finally, a more robust sensor (e.g., an infrared sensor) should be evaluated with this system to determine its utility in relative navigation for AAR. Without adding significant optical filters, automatic f-stop features, and potentially costly image processing, the EO sensor is ineffective in certain conditions, such as low sun angle, night, and in the presence of clouds.

Appendix A. C-12 Model Feature Description

The tanker model for the C-12C was created by the Cyclops test team [21] and contains 29 measured feature locations. It was created using a surveyed area with multiple manual measurements. Figure A.1 shows a picture of the C-12C from a typical refueling viewpoint along with the measured features for the tanker model. The feature descriptions can be found in Table A.1.
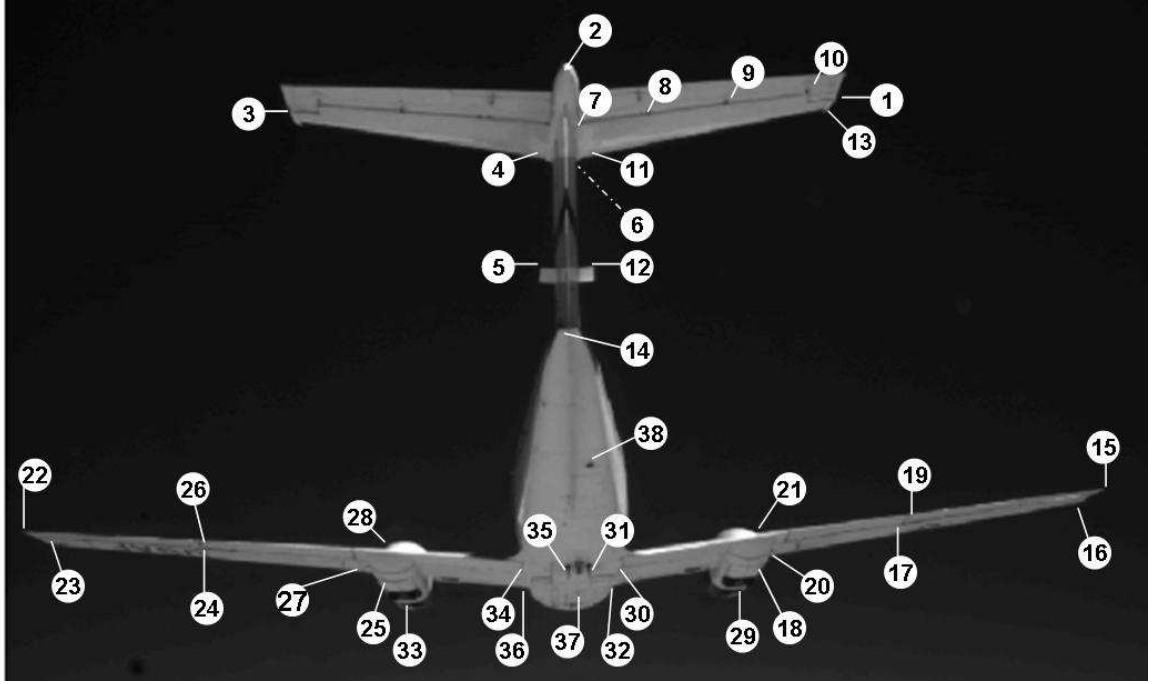

Figure A.1:   The location of the feature points for the C-12C feature model.

Table A.1:   C-12 Model Feature Description.

Point Number	Point Description
1	Right Stab Outboard Gap Tip
2	Aft Tip of Tail Stinger (Light)
3	Left Stab Outboard Gap Tip
4	Left Stab Inboard Aft Corner of Deice Boot
5	Left VOR Antenna Outboard Aft Corner
6	Forward Boom Tip of Tail Section
7	Right Stab Inboard Corner
8	Right Stab Inboard Hinge
9	Right Stab Outboard Hinge
10	Right Stab Outboard Corner
11	Right Stab Inboard Aft Corner of Deice Boot
12	Right VOR Antenna Outboard Aft Corner
13	Right Stab Outboard Corner of Deice Boot
14	Tail Light Bottom Edge
15	Right Aileron TE Tip
16	Right Outboard TE Deice Boot
17	Right Aileron Inboard Forward Corner
18	Right Wing Fuel Drain Tip
19	Right Outboard Flap TE
20	Right Outboard Deice Boot Inboard Corner
21	Right Engine Exhaust
22	Left Aileron TE Tip
23	Left Outboard TE Deice Boot
24	Left Aileron Inboard Forward Corner
25	Left Wing Fuel Drain Tip
26	Left Outboard Flap TE
27	Left Outboard Deice Boot Inboard Corner
28	Left Engine Exhaust
29	Right Engine Black & White Corner
30	Right Wing Root Fwd LE IB Flap Corner
31	Right Black Antenna LE
32	Right IB Deice Boot IB TE corner
33	Left Engine Black & White Corner
34	Left Wing Root Fwd LE IB Flap Corner
35	Left Rock Guard LE
36	Left IB Deice Boot IB TE corner
37	Lower VHF Blade/Fuselage Corner
38	Rear Belly Right Side Vent LE

Bibliography

1. Blackman, Samual. *Multiple-Target Tracking with Radar Application*. Artech House, Norwood, MA, 1986. ISBN 0-89006-179-3.

2. Blackman, Samual and Robert Popoli. *Design and Analysis of Modern Tracking Systems*. Artech House, Norwood, MA, 1999. ISBN 1-58053-006-0.

3. Boeing. *Automated Aerial Refueling Precision Navigation System Design Program - Phase I*. In-House BOEING-STL 2005P0042, Boeing, Boeing, St. Louis, MO, June 2005.

4. DARPA. Http://www.darpa.mil/j-ucas/.

5. DARPA. "DARPA Performs World's First Hands-off Autonomous Air Refueling Engagement". News Release, September 2006.

6. DeMenthon, Daniel and Larry Davis. "Model-Based Object Pose in 25 Lines of Code". *International Journal of Computer Vision*, 15(1-2):123–141, June 2005.

7. Department of the Air Force. "The U.S. Air Force Remotely Piloted Aircraft and Unmanned Aerial Vehicle Strategic Vision", 2005.

8. Doebbler, James, John Valasek, Mark Monda, and Hanspeter Schaub. "Boom and Receptacle Autonomous Air Refueling Using a Visual Pressure Snake Optical Sensor". *AIAA-2006-6504, AIAA Atmospheric Flight Mechanics Conference and Exhibit*. AIAA, Keystone, CO, August 2006.

9. Gonzalez, Rafael C. and Paul Wintz. *Digital Image Processing*, chapter 2, 40–51. Addison-Wesley Publishing Co., 2nd edition, November 1987.

10. Hansen, Joseph, N. Nabaa, G. Romrell, R. Andersen, L. Myers, and Lt. Col J. McCormick. "DARPA Autonomous Airborne Refueling Demonstration Program with Initial Results". ION, Fort Worth, TX, September 2006.

11. Harris, Chris and Mike Stephens. "A Combined Corner and Edge Detector". *Fourth Alvey Vision Conference*, 147–151. 1988.

12. Inc., Boeing SVS. 411 The 25 Way, NE Suite 350, Albuquerque, NM 87109.

13. Johnson, Richard and Dean Wichern. *Applied Multivariate Statistical Analysis*. Prentice Hall, 2002.

14. Lu, C-P, G. Hager, and E. Mjolsness. "Fast and Globally Convergent Pose Estimation From Video Images". *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 22:610–622, 2000.

15. Moravec, Hans. *Obstacle Avoidance and Navigation in the Real World by a Seeing Robot Rover*. Technical report, Carnegie-Mellon University, Robotics Institute, September 1980.

16. Noble, Alison. *Descriptions of Image Surfaces*. Ph.D. thesis, Oxford University, 1989.

17. Office of the Secretary of Defense. "Unmanned Aircraft Systems Roadmap, 2005-2030", August 2005.

18. Pollini, Lorenzo, Mario Innocenti, and Roberto Mati. "Vision Algorithms for Formation Flight and Aerial Refueling with Optimal Marker Labeling". *AIAA-2005-6010, AIAA Modeling and Simulation Technologies Conference and Exhibit*. AIAA, San Fransisco, CA, August 2005.

19. Shen, Fei and Han Wang. "A Local Edge Detector Used for Finding Corners". *Third International Conference on Information, Communications and Signal Processing*. 2001.

20. Smith, S.M. and J.M. Brady. "SUSAN - A New Approach to Low Level Image Processing". *International Journal of Computer Vision*, 23(1):45–78, 1997.

21. Spencer, James, John Bush, Justin Hsia, Karl Kinsler, David Petrucci, and Eric Rucker. *Optical Tracking for Automated Aerial Refueling (AAR)*. Technical Report AFFTC-TIM-06-08, USAF Test Pilot School, Edwards AFB, CA, December 2006.

22. Trucco, Emanuele and Alessandro Verri. *Introductory Techniques for 3-D Computer Vision*. Prentice Hall, Upper Saddle River, NJ, 1998. ISBN 0-13-261108-2.

23. Valasek, John, Jennifer Kimmett, Declan Hughes, Kiran Gunnam, and John L. Junkins. "Vision Based Sensor and Navigation System for Autonomous Aerial Refueling". *AIAA-2002-3441, Proceedings of the First AIAA Conference and Workshop on Unmanned Aerospace Vehicles, Systems, Technologies, and Operations*. AIAA, Portsmouth, VA, May 2002.

24. Vendra, Soujanya. *Addressing Corner Detection Issues for Machine Vision based UAV Aerial Refueling*. Msae, West Virginia University, Morgantown, West Virginia, 2006.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704–0188

1. REPORT DATE *(DD–MM–YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From — To)*
22–03–2007	Master's Thesis	Jul 2005 — Mar 2007

4. TITLE AND SUBTITLE

Optical Tracking for Relative Positioning
in Automated Aerial Refueling

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

James H. Spencer, Maj, USAF

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Air Force Institute of Technology
Graduate School of Engineering and Management (AFIT/EN)
2950 Hobson Way, Bldg 640
WPAFB OH 45433-7765

8. PERFORMING ORGANIZATION REPORT NUMBER

AFIT/GE/ENG/07-22

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Air Force Research Lab, AFRL/VACC
POC: Jacob Hinchman, Jacob.Hinchman@wpafb.af.mil
2130 8th St. Rm 270
Wright-Patterson AFB, OH 45433
(937)255-8427

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION / AVAILABILITY STATEMENT

Approval for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

An algorithm is designed to extract features from video of an air refueling tanker for use in determining the precise relative position of a receiver aircraft. The algorithm is based on receiving a known estimate of the tanker aircrafts position and attitude. The algorithm then uses a known feature model of the tanker to predict the location of those features on a video frame. A corner detector is used to extract features from the video. The measured corners are then associated with known features and tracked from frame to frame. For each frame, the associated features are used to calculate three dimensional pointing vectors to the features of the tanker. These vectors are passed to a navigation algorithm which uses extended Kalman filters, as well as data-linked INS data to solve for the relative position of the tanker. The algorithms were tested using data from a flight test accomplished by the USAF Test Pilot School using a C-12C as a simulated tanker and a Learjet LJ-24 as the simulated receiver. The system was able to provide at least a dozen useful measurements per frame, with and without projection error.

15. SUBJECT TERMS

Automated Aerial Refueling, Optical Tracking, Machine Vision, Unmanned Aerial Vehicle, Project Cyclops, Automated Formation Flight Test

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Lt Col Matthew Goda
U	U	U	UU	109	19b. TELEPHONE NUMBER *(include area code)* (937) 904–9819; Matthew.Goda@wpafb.af.mil

Standard Form 298 (Rev. 8–98)
Prescribed by ANSI Std. Z39.18